MOVIE RECOMMENDATION SYSTEM USING COLLABORATIVE FILTERING

A Dissertation

Submitted

In Partial Fulfillment of the Requirements

For the Degree of

MASTER OF TECHNOLOGY

In

Computer Science & Engineering

Submitted by:

Ismat Anjum

(2001621005)

Under the Supervision of:

Anwar Ahmad Shaikh

(Assistant Professor)

Department of Computer Science & Engineering

Faculty of Engineering

INTEGRAL UNIVERSITY, LUCKNOW, INDIA

July, 2022

CERTIFICATE

This is to certify that **Mrs. Ismat Anjum** (Enroll. No.2001621005) has carried out the research work presented in the thesis titled **"Movie Recommendation System Using cCollaborative Filtering"** submitted for partial fulfillment for the award of the Master of Technology In Computer Science & Engineering from **Integral University, Lucknow** under my supervision.

It is also certified that:

(i) This dissertation embodies the original work of the candidate and has not been earlier submitted elsewhere for the award of any degree/diploma/certificate.

(ii) The candidate has worked under my supervision for the prescribed period.

(iii) The dissertation fulfills the requirements of the norms and standards prescribed by the University Grants Commission and Integral University, Lucknow, India.

(iv) No published work (figure, data, table etc) has been reproduced in the dissertation without express permission of the copyright owner(s).

Therefore, I deem this work fit and recommend for submission for the award of the aforesaid degree.

Mr. Anwar Ahmed Sheikh

(Assistant Professor)

Department of CSE,

Integral University, Lucknow

Date:

Place: Lucknow

DECLARATION

I hereby declare that the dissertation titled **"Movie Recommendation System**: **Using Collaborative filtering"** is an authentic record of the research work carried out by me under the supervision of Mr. Anwar Ahmed Sheikh, Department of Computer Science & Engineering , for the period from August,2021 to May , 2022 at Integral University, Lucknow. No part of this dissertation has been presented elsewhere for any other degree or diploma earlier.

I declare that I have faithfully acknowledged and referred to the works of other researchers wherever their published works have been cited in the dissertation. I further certify that I have not willfully taken other's work, para, text, data, results, tables, figures etc. reported in the journals, books, magazines, reports, dissertations, theses, etc., or available at web-sites without their permission, and have not included those in this M.Tech dissertation citing as my own work.

Date:

Signature
Ismat Anjum
Enrollment. No.**2001621005**

RECOMMENDATION

On the basis of the declaration submitted by "**Ismat Anjum**", a student of M.Tech CSE (FT) , successful completion of Pre presentation on 24/06/2022 and the certificate issued by the supervisor "Mr. Anwar Ahmed Sheikh ", Assistant Professor Computer Science and Engineering Department, Integral University, the work entitled "Movie Recommendation System Using Collaborative Filtering", submitted to department of CSE, in partial fulfillment of the requirement for award of the degree of Master of Technology in Computer Science & Engineering, is recommended for examination.

Program Coordinator Signature	HOD Signature
Dr. Faiyaz Ahmad	**Dr. Kavita Agrwal**
Dept. of Computer Sc. & Engg.	Dept. of Computer Sc. & Engg.
Date:	Date:

COPYRIGHT TRANSFER CERTIFICATE

Title of the Dissertation: **Movie Recommendation System using collaborative filtering**

Candidate Name: **Ismat Anjum**

The undersigned hereby assigns to Integral University all rights under copyright that may exist in and for the above dissertation, authored by the undersigned and submitted to the University for the Award of the M.Tech degree.

The Candidate may reproduce or authorize others to reproduce material extracted verbatim from the dissertation or derivative of the dissertation for personal and/or publication purpose(s) provided that the source and the University's copyright notices are indicated.

ISMAT ANJUM

ACKNOWLEDGEMENT

I am highly grateful to the Head of Department of Computer Science and Engineering for giving me proper guidance and advice and facility for the successful completion of my dissertation.

It gives me a great pleasure to express my deep sense of gratitude and indebtedness to my guide **Mr. Anwar Ahmed Sheikh, Assistant Professor, Department of Computer Science and Engineering,** for his valuable support and encouraging mentality throughout the project. I am highly obliged to him for providing me this opportunity to carry out the ideas and work during my project period and helping me to gain the successful completion of my Project.

I am also highly obliged to the Head of Department, **Dr. Kavita Agarwal, Department of Computer Science and Engineering** and PG Program Coordinator **Dr. Mohd Faiyaz Ahmad, Assistant Professor, Department of Computer Science andEngineering,** for providing me all the facilities in all activities and for his support and valuable encouragement throughout my project.

My special thanks are going to all the faculties for encouraging me constantly to work hard in this project. I pay my respect and love to my parents and all other family members and friends for their help and encouragement throughout this course of project work.

Date:

Place: Lucknow **ISMAT ANJUM**

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF SYMBOLS & ABBREVIATIONS

IFS	-	Information Filtering System
RS	-	Recommendation System
UI	-	User Interface
CF	-	Collaborative Filtering
KNN	-	K- Nearest Neighbour
ICF	-	Item Based Collaborative Filtering
AHP	-	Analytic Hierarchy Process

ABSTRACT

A movie recommendation is essential in our social life since it has the ability to provide more enjoyment than other forms of entertainment. Depending on the users' interests or the popularity of the films, a system like this may provide them with a selection of movies to watch. A recommendation system is used for the purpose of suggesting products to purchase or to view. In the meanwhile, consumers cannot enjoy all accessible new releases or unseen movies owing to their restricted time. They also still need to pick which movies to view when they have extra time. This scenario is not favourable for the movie sector too. It is essential to provide the user with movie recommendations so that the user does not have to spend a significant amount of time searching for content that they would like. As a result, the function of the movie recommendation system is quite important in order to acquire user-specific movie choices. After doing considerable research on the internet and consulting a large number of scholarly articles, we came to the conclusion that the suggestions generated by Collaborative Filtering only use a single method for converting text to vectors and only use a single method for determining the degree to which vectors are similar to one another. Our project's goal is to create a recommendation engine that responds to the user in order to obtain ideas for a movie.In order to satisfy consumers in picking what movies to watch and to improve movie sales, a system that can recommend relevant movies is necessary, either unseen in the past or recent releases. This study focuses on the review on hybrid technique, a blend of content-based and collaborative filtering, utilising a new perspective.

Keywords Movie recommendation, Filtering method, Hybrid Method

CHAPTER: 1

INTRODUCTION

1.1.INTRODUCTION

The objective of this chapter is to discuss about Information Filtering System, different type of Recommendation system, motivation,methodology, contribution, and outline of the Project.

Information Filtering System (IFS) is a system that removes redundant or unwanted information from an information stream using automated or computerized method. An information system assist user by filtering the data source and deliver relevant information to the user. When the delivered information comes in the form of suggestion, then it is called as Recommender System. IFS is divided into three parts based on E-commerce, as shown in Fig. 1.1.

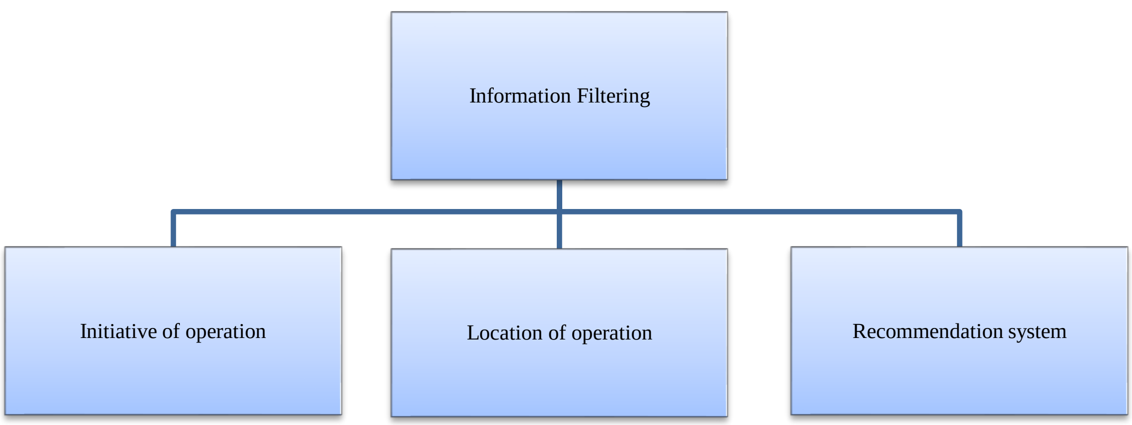


Fig. 1.1: Classification of Information Filtering System

Information filtering systems are classified into following types:

a. Initiative of operation: Distinguishing between active and passive IF Systems.
b. Location of operation: Distinguishing between systems located at the information source, filtering servers, and user sites.

c. Recommendation System (RS) : The evolution of recommendation system starts as follows:

(1) Recommender Systems is a special type of information filtering technique.

(2) Recommendation Systems provide technology that helps users in finding relevant content on internet.

(3) Recommendation Systems assist user in overcoming information on internet that is not correct information for user.

(4) Recommendation Systems are directly involved in assigning users to make decision and satisfy them into their current need.

(5) Recommendation Systems collect information on the preference of its users for a set of items (e.g. movies, songs, books etc.) [1].

(6) Recommendation Systems can be defined as navigating information in an efficient and satisfying way.

(7) Recommendation Systems is an automated technique (mechanism) to seek out relevant as well as new information.

All the above definition shows the characteristics of recommendation system that how recommendation system involves the user/client on different areas.

The first recommender system was discussed in the work of Rensick and Varian (1997) and since then they are evolving continuously to achieve higher degree of accuracy as well as user satisfaction. After the existence of recommendation system, other systems like Letizia (Lieberman 1995) and Fab (Balabanovic and Shoham 1997) faded out with passage of time. Letizia, invented by Letizia Alvarez de Toledo, is a User Interface (UI) agent that assists a user while browsing on internet, as shown in Fig.1.2. Letizia is uses by Mosaic and Netscape.

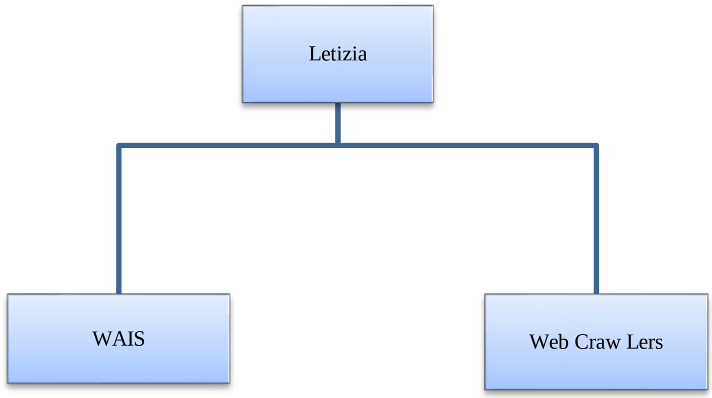


Fig.1.2: Types of Letizia

Fab is a distributed implementation of a hybrid system and is part of the **Stanford University** digital library project.

The tree structure of different types Recommendation System is given in Fig.1.3. In this figure, we categorize Recommendation System in three parts, i.e. Content Based RS, Collaborative Filtering Technique, Hybrid Filtering Technique. In our Project, we mainly focus on Collaborative Filtering Technique In general recommendation system has two types, i.e., (i) Non-Personalize and (ii) Personalize.

Non-personalize recommendation comes into the field of advertisement. Personalized is that RS in which we collect the information of user to recommend item to his/her interest.

1.2. TYPES OF RECOMMENDATION SYSTEM

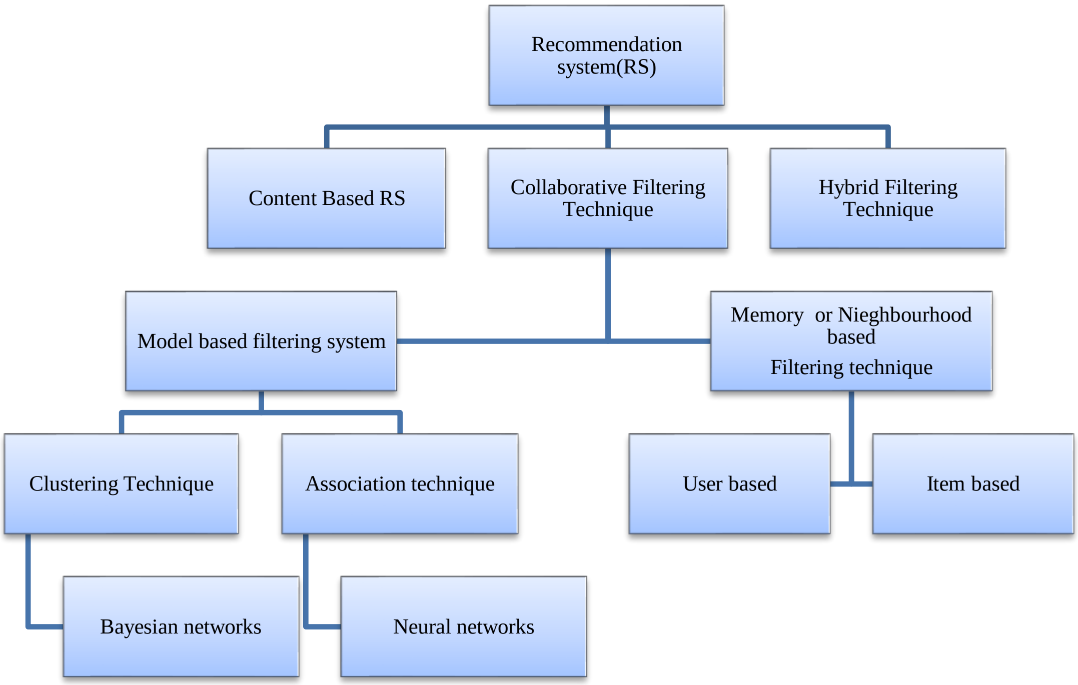


Fig.1.3: Tree structure of types of recommendation system

1.2.1 Content Based Recommendation System:

Content based recommendation system recommends item based on comparison between the content of the item and a user profile. The user will be recommended items similar to the ones the user preferred in the past. A content based recommender works with data that the user provides, either explicitly such as rating or implicitly such as clicking on a link based on that data, a user profile is generated, which is used to make suggestion to the user [2].

1.2.2. Collaborative Filtering System:

Collaborative filtering (CF) is a method of making automatic prediction (filtering) about the interests of a user by collecting preferences or taste information from many users, in collaborative way.

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

1.2.2.1. 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.

1.2.2.2. Item-based filtering:

Unlike the user-based filtering method, itembased 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.

1.2.3. Hybrid Recommendation System:

Recent research has demonstrated that a hybrid approach, combining collaborative filtering and content-based filtering could be more effective in some cases. Hybrid approaches can be implemented in many ways: by making content-based and collaborative-based predictions separately and then combining them by adding content-based capabilities to a collaborative-based approach (and vice versa). Several studies empirically compare the performance of the hybrid with the pure collaborative and content-based methods and demonstrate that the hybrid methods can provide more accurate recommendations than pure approaches. These methods can also be used to overcome some of the common problems in recommender systems such as cold start and the sparsity problem.

Netflix is a good example of the use of hybrid recommender systems. The website makes recommendations by comparing the watching and searching habits of similar users (i.e. collaborative filtering) as well as by offering movies that share characteristics with films that a user has rated highly (content-based filtering).

1.3. ADVANTAGE OF MOVIE RECOMMENDATION SYSTEM:

- Utilising this data to recommend the most popular movies to user based on their star ratings, cloud increase their content consumption.
- The popularity-based recommendation system eliminates the need for knowing other factors like user browsing history, user preferences, the star cast of the movie, genre, and other factors.

1.4. DISADVANTAGE OF MOVIE RECOMMENDATION SYSTEM:

- Recommendations are not personalized as per user attributes and all users see the same recommendations irrespective of their preferences
- Another problem is that the number of reviews (which reflects the number of people who have viewed the movie) will vary for each movie and hence the average star rating will have discrepancies.

- The system doesn’t take into account the regional and language preferences and might recommend movies in languages that a regional dialect speaking individual might not understand.
- A popularity based recommendation system when tweaked as per the needs, audience, and business requirement, it becomes a hybrid recommendation system. Additional logic is added to include customization as per the business needs.

1.5. MOTIVATION:

Due to rapid growth in web technologies every intellectual person on internet are facing a huge amount of information to navigate that information in which they are interested in an efficient and satisfying way. The dependency of people on internet made very difficult for people to find information that is relevant to their needs and interest. To handle this problem, recommendation system came into existence to offering a user for an automated mechanism to seek out relevant as well as new information. It is of great importance for the success of E-commerce and IT industry in these days and gradually gains popularity in various applications (e.g. YouTube, Google, and amazon). When we were searching videos on YouTube there we find a section of recommended video, then we have eagerness in our mind that how it works.

1.6. BACKGROUND:

In the early time we use the content based recommendation system, and in this system we only recommend the item according to user’s browsing history and we can only recommend item to a single user. And when we want to recommend a single item among multiple users then this system is not suitable to do this task. Collaborative filtering system came into existence to full fill these types of works.

In the newer, narrower sense, collaborative filtering is a method of making automatic predictions (filtering) about the interests of a user by collecting explicit rating or taste information from many users (collaborating). The underlying assumption of the collaborative filtering approach is that if a person X has the same opinion as a person *Y* on an issue, X is more likely to have Y's opinion on a different issue than that of a randomly chosen person. For example, a collaborative filtering recommendation system for television tastes could make predictions about which television show a user should like given a partial list of that user's tastes (likes or dislikes). Note that these predictions are specific to the user, but use information obtained from many users. This differs from the simpler approach of giving an average (non-specific) score for each item of interest, for example based on its number of votes.

Collaborative filtering system is a recommender system that recommends items based on similarity measures between users and/or items .The items recommended to a user are those preferred by similar users. This sort of recommendation system can use the ground work laid. Collaborative system are usually categorised into three groups: User Based, Item Based and Model based but we mainly focused on user based and item based. User-based CF methods identify users that are similar to the queried user, and estimate the desired rating to be the average ratings of these similar users. Similarly, item-based CF identify items that are similar to the queried item and estimate the desired rating to be the average of the ratings of these similar items [3].

1.7. SCOPE:

The objective of this project is to provide accurate movie recommendations to users. The goal of the project is to improve the quality of movie recommendation system, such as accuracy, quality and scalability of system than the pure approaches. This is done using Hybrid approach by combining content based filtering and collaborative filtering, To eradicate the overload of the data, recommendation system is used as information filtering tool in social networking sites .Hence, there is a huge scope of exploration in this field for improving scalability, accuracy and quality of movie recommendation systems Movie Recommendation system is very powerful and important system. But, due to the problems associated with pure collaborative approach, movie recommendation systems also suffers with poor recommendation quality and scalability issues.

1.8. METHODOLOGY:

A Research procedure comprises of arrangement of systems or steps important to effectively complete research and the favored sequencing of these means to create new information, or to offer another way of tolerating present algorithm used. This is a novel approach towards the study. In distinct phases the work has been designed, each stage having its own importance. We started our work by a state of the art. After identification of the problem, we have done a literature survey in a detailed,about movie recommendation system by using various approaches. This literature review was followed by analysis of the algorithm used before. The results were gathered, analyzed and conclusions were drawn based on the results obtained from the review.

STAGE I: PROBLEM IDENTIFICATION AND ALGORITHMSELECTION

The selection of the proper problem and its identification at the outset is the most important phase. After a detailed study of the different algorithms used for movie recommendation system, we found that there is problem of sparcity and cold start for new users ,to overcome these problem we used the concept of pearon correlation and cosine similarity.

STAGE II: LITERATURE STUDY

A review of the state of the art was made after the identification of the problem. With expertise the sparcity and cold start issues, the most important thing is to understand its basics. A literature survey was conducted in order to gain the solid background for the analysis of various algorithm used before for cold start and sparcity. Different algorithms and their functionalities were studied. In fact, this literature study enabled us to understand in detail how cold start and sparcity can be avoided. Also, we found that how users can get best recommendation for movies.

STAGE III: BUILDING FRAMEWORK:

In order to analyze the problem, we take some of the best algorithms and compare them with each other in terms of their efficiency. After that, we have proposed a novel approach using algorithm that can be used to increase the efficiency, and which will work better than the previous algorithms proposed. The results were gathered and analyzed in the fourth step of research design.

STAGE IV: RESULT ANALYSIS:

The most important stage is the last stage. In this phase, the results are obtained by working on the algorithms proposed in our paper and also we compare our algorithms with the previous algorithms used so as to check whether the proposed algorithm works better than the previous one or not.

1.9 Dissertation Outline

The rest of this document is organized as follows.

Chapter 2

In this chapter, I reviewed various national and international journals and publications to identify the real problem statement for doing appropriate research to get better results.

Chapter 3

In this chapter, our proposed work discusses and explained in detail the problem formulation to use Collaborative Filtering System. In order to give more clear view of the implementation details involved part of the code and results are presented as the algoritham,flow chart and graphs.

Chapter 4

In this chapter, the metrics that were used to measure the performance of proposed work along with diagrams that illustrate the performance measurements.

Chapter 5

In this chapter the result and comparative analysis explained in detail. the implementation details and results of the implementation details involved and snapshot of the implementation.

Chapter 6

In this chapter conclusion and some of the future scopes discussed of this work.

CHAPTER: 2

LITERATURE REVIEW

2.1 LITRATURE REVIEW:

A much research work has been made and is still being done on using either collaborative, content based or hybrid filtering methods

In a work by Ahuja et al. (2019), a recommendation strategy that utilises both KNN algorithms and the K-means technique is envisioned. The client is approached in order to obtain information about the finer points. The user's userid, gender, and age are all provided by the user. The pandas module divides the data generally according to the customer and movies into separate dfs in the processing module. For the K-means module, the movie genre can be shown on an edge of data. WCSS determines the appropriate number of clusters. Pearson's correlation similarity and regularisation model uses a matrix to calculate the connection. When determining film ratings, the algorithm employs KNN predictions and the UC grid to compare results. A pre-processing step eliminates outliers in both Indira and Kavithadevi (2019) and the present study (NPCA-HAC). This is followed by the use of feature selection and principal component analysis. K-means and HAC are used to group the selected characteristics. A trust rating algorithm is used to rate the clustered groupings. The clustering approach utilised in this study resulted in a loss of data owing to dimensionality reduction. Prediction performance and scalability are mutually exclusive. As a result of collaborative filtering, data sparsity, excessive computing complexity, and over-specification can be reduced. Combination models are suggested to provide a real-time item that is tailored to the needs of the consumers. Final recommendation list categorization is based on the MP neuron model. Scalability is an issue that has not been addressed in the suggested paradigm. The new item-centered strategy employs CF and CBF techniques and proposes items based on feelings. Reviews and comments on a certain product are used to extract feelings. Emotions can be used to produce item-to-item similarities. It's a good paradigm, however it doesn't take into consideration scalability and computing time. The method of discovering and crafting a film by taking into account the cinema formats of potential audiences. Users are grouped together based on their shared tastes and the ratings they have given to films they have seen. RNN may be used to evaluate and create movies, as well as to discover patterns in the viewing habits of similar groups of users.

Three methods are employed in [3] and in this paper: a basic RS, a content-based approach, and a CF approach. Machine learning is employed in this project. The chart for the basic recommender system is made using IMDB's method for weighted rating. Two further techniques are followed. Sparsity, new user problems, and decreasing computing efficiency all contribute to decreased performance. It has been shown that item-based collaborative filtering (ICF) is superior to user-based CF in terms of analysis and data processing complexity, as demonstrated in this work. Working performance may be improved by utilising item content and feature vectors. A sign-up system collects the user's personalised information. The experiment's results are used to determine the degree of intimacy between participants. The adjacency matrix of user proximity is formulated at the end of the trial. This paper (Xu X, 2018) presents a methodology that may take into account feedback from both the item and the user community. It employs Ml tools to increase the quality of suggestion in order to strengthen the model's deep learning. Mapped users and things create a representation of the person and the item. Items may be retrieved and ranked using this visual depiction. As a result, the issue is seen as a way to sort things out. To hone the framework, back propagation is employed. Two collaborative models are described by Wu et al. (2019) for the usage of a recommender system. User and item collaborative model strategies are used in this work to design a system that takes use of commonalities across entities Explicit rating refers to how customers rate an item on a certain scale. We can calculate the total number of NN for each user.

PCS [2] is used to discover the correlation between user ratings. Rather of focusing on what the item's users enjoy, items focus on what the thing likes. Recommendation is made based on the item's similarity to the target [6].

Sang-Min Choi, et. al. [18] mentioned about the shortcomings of collaborative filtering approach like sparsity problem or the cold-start problem. In order to avoid this issue, the authors have proposed a solution to use category information. The authors have proposed a movie recommendation system which is based on genre correlations. The authors stated that the category information is present for the newly created content. Thus, even if the new content does not have enough ratings or enough views, still it can pop up in the recommendations list with the help of category or genre information. The proposed solution is unbiased over the highly rated most watched content and new content which is not watched a lot. Hence, even a new movie can be recommended by the recommendation system.

George Lekakos, et. al. [19] proposed a solution of movie recommendation using hybrid approach. The authors stated that Content based filtering and Collaborative filtering have their own shortcomings are can be used in a specific situation. Hence, the authors have come up with a hybrid approach which takes into consideration both content-based filtering as well as collaborative filtering. The solution is implemented in 'MoRe' which is a movie recommendation system. For the sake of pure collaborative filtering, Pearson correlation coefficient has not been used. Instead, a new formula has been used. But this formula has an issue of 'divide by zero' error. This error occurs when the users have given same rating to the movies. Hence, the authors have ignored such users. In case of pure content-based recommendation system, the authors have used cosine similarity by taking into consideration movie writers, cast, directors, producers and the movie genre. The authors have implemented a hybrid recommendation method by using 2 variations - 'substitute' and 'switching'.

Both of these approaches show results based on collaborative filtering and show recommendations based on content-based filtering when a certain criterion is met. Hence, the authors use collaborative filtering technique as their main approach.

Debashis Das, et. al**.** [20] wrote about the different types of recommendation systems and their general information. This was a survey paper on recommendation systems. The authors mentioned about Personalized recommendation systems as well as non-personalized systems. User based collaborative filtering and item based collaborative filtering was explained with a very good example. The authors have also mentioned about the merits and demerits of different recommendation systems.

Prince Praveen [21] proposed a Movie Recommendation system is a system that provides movie suggestions to users based on some dataset. Such a system will predict what movies a user will like based on the attributes of previously liked movies by that user. Content-Based recommendations have long been in fashion but they tend to overlook some great suggestions that may not be covered by mere cosine similarities. To overcome such shortcomings, we will combine collaborative filtering techniques having a User-User matrix with neural networks to provide users(who have already rated movies previously) with appropriate suggestions.

The recommendation system is a component of everyday life where individuals rely on knowledge to make decisions about what they want to do [14]. Collaboration filtering models take into account a user's prior purchases, as well as the judgments made by other users who have made comparable purchases or given numerical ratings to the things they purchased. After that, several models are employed to predict what the user would be interested in (or how they rate certain goods). However, despite the fact that several approaches have been established in the past. Although search is still used in many apps, which customise recommendations and cope with a lack of accuracy, it is still being utilised because of its widespread use. These demands pose a few difficulties. Alternating

LeastSquares, Singular Value decomposition, K-Nearest Neighbor method, and Normal predictor algorithm have been utilised by various academics to address this problem. Memory-based and model-based collaborative filtering approaches are the two main types. Methods relying on memory may be simply adapted to use all the ratings before the filtering phase, thereby ensuring that their findings are always up to date. On the other hand, a model-based system such as a neuralnetwork, develops a model that learns from the knowledge of user-item evaluations and recommends new goods. In order to produce a stronger and more accurate recommendation system, the recommender system still has to be improved. As a result of the system's recommendations, customers may learn more about products that may be of interest to them. In this study, a variety of approaches are discussed. The needs of life are never enough to satisfy a person's self-satisfaction, and so is the constant need for enjoyment in daily life. Watching movies is one of the fun things to do in your spare time. Movies are universally popular, regardless of the genre or the age of moviegoers. This is why the movie industry is so lucrative [11].

Many films or movies are released at the same time in order to satisfy the audience and make money. However, some people, because to time or money constraints, are unable to see all of the new releases. Some people prefer to view movies at a later time, and this might lead to them forgetting what they were supposed to see. To jog their memory about what they wish to see, most consumers turn to the Internet, such as online retailers selling or renting movies [10].

Streaming video-ondemand services are now readily available on the web and on smart phones, thanks to the use of certain video-streaming applications. Smart televisions and set-top boxes with video-streaming capabilities are becoming more commonplace nowadays. Categorization methods that employ a variety of data organization and classification methodologies are common in the field of machine learning. Data for training classifiers is possible [8].

The dependency of people on internet made very difficult for people to find information that is relevant to their needs and interest. To handle this problem recommendation system came into existence for offering a user for an automated mechanism to seek out relevant as well as new information. It is of great importance for the success of E-commerce and IT industry in these days and gradually gains popularity in various applications (e.g. YouTube, Google, and amazon). When we are searching videos on YouTube there we find a section of recommended video, then we have eagerness in our mind that how it works.

When we are browsing on internet we face a lot of information may be some of our need and some are not. And when we are searching for some items on Amazon like laptop or mouse and after sometime if we switch onto Google search engine there we found that there are advertisements of items that we are searched on Amazon. We wondered how it can be possible and later on we came to know about this. It is done by machine learning and data mining concepts which leads us to recommendation system. Machine learning is a technique in which the system learns or reads our previous activities and the matching patterns of the user data on the behalf of this our system recommends us to do some task.

The key reason why many people seem to care about recommender systems is money. For companies such as Amazon, Netflix, and Specify, recommender systems drive significant engagement and revenue. But this is the more cynical view of things. The reason these companies (and others) see increased revenue is because they deliver actual value to their customers recommender systems provide a scalable way of personalizing content for users in scenarios with many items.

Another reason why data scientists specifically should care about recommender systems is that it is a true data science problem. That is, at least according to my favorite definition of data science as the intersection between software engineering, machine learning, and statistics. As we will see, building successful recommender systems requires all of these skills.

An Information Filtering Technology, commonly used on E-commerce websites that uses a collaborative filtering to present information on items and products that are likely to be of interest to the user. In present the recommender system will use details of the registered user's profile and opinions and habit of their whole community of users and compare one information to reference characteristics to present the recommendation.

There is an extensive class of Web applications that involve predicting user responses to options. Such a facility is called a recommendation system. We shall begin this chapter with a survey of the most important examples of these systems. However, to bring the problem into focus, two good examples of recommendation systems are:

- Offering news articles to on-line newspaper readers, based on a prediction of reader interests.
- Offering customers of on-line retailer suggestions about what they might like to buy based on their past history of purchases and/or product searches.

By Ching-Seh (Mike) Wu,Deepti Garg,Unnathi Bhandary [23].Proposed 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.

CHAPTER:3

PROPOSED METHODOLOGY

3.1 PROPOSED METHODOLOGY:

In this chapter we will build Movie recommendation systems with various approaches and with each step, we will get more advanced and improve the quality of the suggestions made by the proposed system.

3.1.1. Content-Based Recommendation System :

This approach for recommending movies does not involve other users. Based on what we like, our algorithm will pick similar items i.e items having similar content and recommend us.

In this approach, the diversity in recommendations will be the least as it only takes into consideration what the user specifically likes. E.g, A user that says they like Action movies will only be recommended other action movies until they try some other genre autonomously and decide to give it a like. Ofcourse, there are many categories we can calculate the similarity on: as in our case of movies, we can decide to find similarity based on genre, keyword, cast,director and so on.

Algorithm used :

3.1.1.1.COSINE SIMILARITY

To find similar content for our item, we used the cosine similarity algorithm. The dot product between two vectors is equal to the projection of one of them on the other. Therefore the dot product of two identical vectors is equal to their squared modules. On the other hand if the two vectors do not share any directions, the product will be zero. General formula for calculating dot product is given below:

This dot product is important when defining the similarity as it is directly connected to it. The definition of similarity between two vectors u and v is in fact the ratio between their dot products and product of their magnitudes.

Thus, this will be equal to 1 if the two vectors are identical or it will be 0 if the two are orthogonal.

3.1.2. Collaborative Filtering Recommendation System :

The previous approach didnt involve other users and in so it had some shortcomings. Such limitations involve the recommendations not being diverse as discussed before. To solve such problems we use the collaborative filtering technique. This approach is based on the idea that the user rates, and the system will recommend different movies that the user has not watched but the other users similar to our test user have watched and liked. This type of collaborative filtering approach is called the User-to-User Collaborative filtering approach as we find similar users to our user.

To determine whether the two users are similar or not, we consider the movies watched by both of them and how they rated them. Thus by looking at items in common, we will predict the ratings a user will give to a movie who hasnt watched it yet, based on its similar user rates.

Algorithms Used:

3.1.2.1. K Nearest Neighbors:

The standard method of Collaborative Filtering is known as Nearest Neighborhood algorithm. We have an n Ã— m matrix of ratings, with user u, i = 1, …n and item p, j=1, m. Now we want to predict the rating r if target user i did not watch/rate an item j. The process is to calculate the similarities between target user i and all other users, select the top X similar users, and take the weighted average of ratings from these X users with similarities as weights.

However, not all users have the same baseline for giving ratings to movies. Some users may tend to give high scores generally while some are pretty strict with their ratings even though they are satisfied with the items. To avoid such bias, we will subtract each users average ratings of all the items when computing weighted average, and add it back for the target user as shown:

3.1.2.1. Matrix Factorization:

Sparsity is a big issue that needs to be addressed while creating collaborative filtering recommendation systems. Our approach creates matrices where rows are unique users in our environment and the columns represent different movies and the values within are the ratings that different users give to movies. However, it is rather obvious that not all movies will be rated by each user. Thus this matrix of ours faces the problem of sparsity that needs to be solved. For this purpose, w use Matrix Factorization. In this method, we decompose the original sparse matrix to low-dimensional matrices with latent features. Therefore matrix factorization gives us how much a user is aligned with a set of latent features, and how much a movie fits into this set of latent features.

The advantage of this approach over the previous algorithm is that even though two users havent rated same movies, it is still possible to find out the similarity between them if they share similar latent features.

Cold Start Problems

Cold start problems can be handled by recommendations based on meta-information, such as:

- For new users, we can use their location, age, gender, browser, and user device to predict recommendations.
- For new movies, we can use genre, cast, and crew to recommend it to target users.

3.2 DATA SET DESCRIPTION:

The dataset used for this project is the well known Movielens dataset (http://grouplens.org/datasets/movielens/100k/) to analyze the behavior of our proposed system. We study the public Movielens dataset to conduct the experiments, which is accessible online, having 100,000 ratings by 943 users or participants on 1682 movies, of scale 1–5. The dataset is divided into 80% training data and 20% test data for verification of the result. Movies are classified into 19 types viz. action, animation, horror, comedy, etc.

For our own system, we'll use the open-source MovieLens dataset from GroupLens. This dataset contains 100K data points of various movies and users.

We will use three columns from the data:

- Userid
- Movieid
- Rating
- You can see a snapshot of the data in figure 3, below:

	userId	movieId	rating
0	1	1	4.0
1	1	3	4.0
2	1	6	4.0
3	1	47	5.0
4	1	50	5.0
...	...	...	...
100831	610	166534	4.0
100832	610	168248	5.0
100833	610	168250	5.0
100834	610	168252	5.0
100835	610	170875	3.0

Figure 3.1: Snapshot of data

3.3. SOFTWARE AND HARDWARE REQUIREMENTS:

3.3.1 Software Requirements

- Python
- Numpy
- Django

3.3.2 Hardware Requirements

- A PC with Windows/Linux OS
- Processor with 1.7-2.4gHz speed
- Minimum of 8gb RAM

3.4 OBJECTIVES

The objective of this chapter is to analyze the CFS for movie recommendation system.

1. The main objective of my work is to overcome some of common problem in recommendation system such as cold start and The sparsity problem by using collaborative filtering recommendation system.
2. They help the user find items of their interest
3. Improving the Accuracy of the recommendation system
4. Improve the Quality of the movie Recommendation system
5. Improving the Scalability.
6. Enhancing the user experience.

3.5 PROBLEM STATEMENT

- 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.
- The goal of a recommendation system is either customer-driven or business-driven. For example, it might be:

- Identify movies that customers want to watch, as demonstrated by their post-viewing rating. (A user-satisfaction metric.)

Or it might be:

- Identify a list of movie recommendations, which contains at least one that the user will start watching as their next selection. (An engagement metric.)

The goal of a recommendation system is either customer-driven or business-driven. For example, it might be:

- Identify movies that customers want to watch, as demonstrated by their post-viewing rating. (A user-satisfaction metric.)

Or it might be:

- Identify a list of movie recommendations, which contains at least one that the user will start watching as their next selection. (An engagement metric.)

Based on our study of the architecture and working model of different recommendation based systems like Mosaic, Netscape, YouTube etc. and we analysed that these systems use the content based recommendation which is inefficient in nature itself and it is replaced by the collaborative filtering system in which we firstly find out the similarity between users and then we predict or recommend an item for a particular user among those similar users which is not the property of content based filtering system, here recommendation is based on the users previous records.

Recommender systems handle the problem of information overload that users normally encounter by providing them with personalized, exclusive content and service recommendations. Recently, various approaches for building recommendation systems have been developed, which can utilize either collaborative filtering, content-based filtering or hybrid filtering. Collaborative filtering technique is the most mature and the most commonly implemented. Collaborative filtering recommends items by identifying other users with similar taste; it uses their opinion to recommend items to the active user. Collaborative recommender systems have been implemented in different application areas. Group Lens is a news-based architecture which employed collaborative methods in assisting users to locate articles from massive news database. Ringo is an online social information filtering system that uses collaborative filtering to build users profile based on their ratings on music albums. Amazon uses topic diversification algorithms to improve its recommendation. The system uses collaborative filtering method to overcome scalability issue by generating a table of similar items offline through the use of item-to-item matrix. The system then recommends other products which are similar online according to the users' purchase history.

And now we are working on the recommendation which is based on posting of any material related to subjects. Suppose a user John posted a document on the topic of artificial intelligence and gives also its tags, labels and comment text and another user suppose Mike also posted a paper on artificial intelligence with some different contents and with same tags and labels. And when a user searches for a paper on artificial intelligence in our blogging site then it shows the paper of John on artificial intelligence and also of Mike and it may recommends of someone else who have posted the same paper on artificial intelligence.

3.6. ALGORITHM USED:

I. K- NEAREST NEIGHBORS

II. COSINE SIMILARITY

K- NEAREST NEIGHBORS

The standard method of Collaborative Filtering is known as Nearest Neighborhood algorithm. We have an n Ã— m matrix of ratings, with user u, i = 1, …n and item p, j=1, m. Now we want to predict the rating r if target user i did not watch/rate an item j. The process is to calculate the similarities between target user i and all other users, select the top X similar users, and take the weighted average of ratings from these X users with similarities as weights.

However, not all users have the same baseline for giving ratings to movies. Some users may tend to give high scores generally while some are pretty strict with their ratings even though they are satisfied with the items. To avoid such bias, we will subtract each users average ratings of all the items when computing weighted average, and add it back for the target user as shown:

In K-NN whole data is classified into training and test sample data. In a classification problem, k nearest algorithm is implemented using the following steps.

1. Pick a value for k, where k is the number of training examples in the feature space.
2. Calculate the distance of unknown data points from all the training examples.
3. Search for the k observations in the training data that are nearest to the measurements of the unknown data point.

4. Calculate the distance between the unknown data point and the training data.
5. The training data which is having the smallest value will be declared as the nearest neighbor.

In the KNN-regression problem, the only difference is that the distance between training points and sample points is evaluated and the point with the lowest average distance is declared as the nearest neighbor. It predicts the result on the basis of the average of the total sum.

COSINE SIMILARITY

To find similar content for our item, we used the cosine similarity algorithm. The dot product between two vectors is equal to the projection of one of them on the other. Therefore the dot product of two identical vectors is equal to their squared modules. On the other hand if the two vectors do not share any directions, the product will be zero. General formula for calculating dot product is given below:

This dot product is important when defining the similarity as it is directly connected to it. The definition of similarity between two vectors u and v is in fact the ratio between their dot products and product of their magnitudes.

Thus, this will be equal to 1 if the two vectors are identical or it will be 0 if the two are orthogonal.

3.6.1. COMPUTING SIMILARITY AMONG USERS:

We use the concept of Pearson correlation, cosine similarity and user based prediction computing formula to find the similarity among users.

3.6.1.1. Pearson Correlation

This method computes the statistical correlation (Pearson's *r*) between two user's common ratings to determine their similarity. The correlation is computed by the following:

$$\text{sim (i,j)}= \frac{\Sigma u\in U(Ru,i-\bar{R}u)(Ru,j-\bar{R}u)}{\sqrt{\Sigma u\in U(Ru,i-\bar{R}u)^2}\sqrt{\Sigma u\in U(Ru,j-\bar{R}u)^2}}.$$

(Eq. 3.1)

Where,

i: Set of items rated by the user.

R_u, i: Is the rating given to item I by user (u).

$\bar{R}_u$: Is the mean rating given by user (u).

3.6.1.2 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. The cosine of 0° is 1, and it is less than 1 for any other angle.

For N-dimensional vector of items, measure two customers A and B

$$\text{Similarity }(\vec{A},\vec{B})= \cos(\vec{A},\vec{B})= \frac{\vec{A}.\vec{B}}{||(\vec{A})||*||(\vec{B})||} \quad \text{(Eq. 3.2)}$$

When computing cosine similarity, one cannot have negative ratings and unrated items are treated as having a rating of zero.

3.6.1.3. User Based Prediction Computing Formula:

$$p_{C,e} = \bar{r}C + \frac{s(C,A)(rA,e - \bar{r}A) + s(C,D)(rD,e - \bar{r}D)}{(|s(C,A)| + |s(C,D)|)} \quad \text{(Eq. 3.3)}$$

(pC,e): User C's prediction for Equilibrium

$\bar{r}C$: Mean rating of user C

$\bar{r}A$: Mean rating of user A

$\bar{r}D$: Mean rating of user D

s(C,A): similar user A and C

s(C,D): similar user C and D

From above formula we find the prediction of user in unknown item. Due to this, then we recommend this item to that user whose was not rated this item. Now we consider a table of ratings of different users.

Table 3.1 Ratings matrix

	Item 1	Item 2	Item 3	Item 4
User A	4	?	3	5
User B	?	5	4	?
User C	5	4	2	?
User D	2	4	?	3
User E	3	4	5	?

Consider the ratings matrix in Table 1, we want to find User C's prediction for Equilibrium (pC,e) with the following configuration:

• Pearson correlation.

• Neighborhood size of 2.

$\bar{r}C = \frac{(5+4+2)}{3}$

$\bar{r}C = 3.667$

C's mean rating is 3.667. There are only two users who have rated Equilibrium, and therefore only two candidate users for the neighborhood:

$$s(C,A) = \frac{(4-4)*(5-3.67)+(3-4)(2-3.67)}{\sqrt{((0)^2}+(-1)^2)*\sqrt{((1.33)^2+(-1.67)^2)}}$$

$$=0.784$$

A and D, *s*(*C*,*A*) = 0.784 and *s*(*C*,*D*) = −0.518 from Equation 1.1.The prediction pC,e is therefore computed as follows:

$$p_{C,e} = \bar{r}C + \frac{s(C,A)(rA,e-\bar{r}A)+s(C,D)(rD,e-\bar{r}D)}{(|s(C,A)|+|s(C,D)|)}$$

$$= 3.67 + \frac{0.784*(5-4)+ -0.518*(2-3)}{0.784+0.518} = 4.667$$

Here we calculated the value of user rating of C. On the behalf of this rating we can recommend items to the user.

3.7. SYSTEM ARCHITECTURE OF PROPOSED SYSTEM:

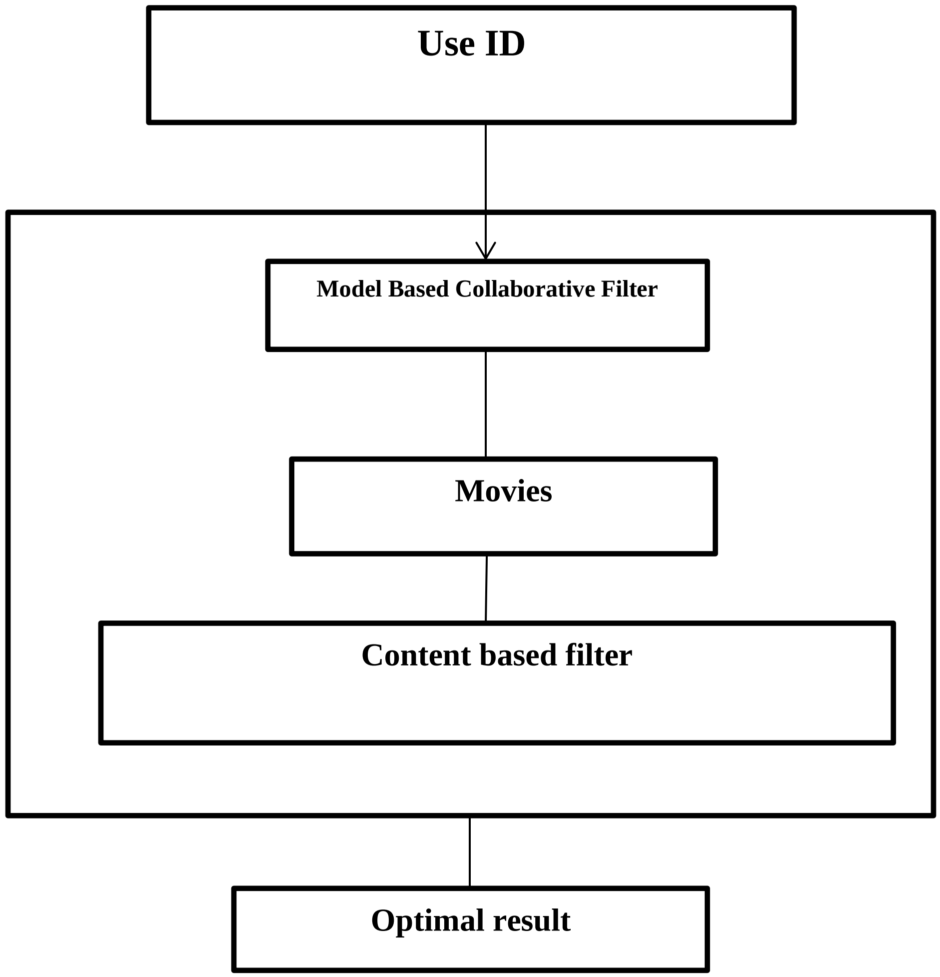


Fig:-3.2 Architecture for hybrid approach

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 work 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.

3.8. DATA FLOW:

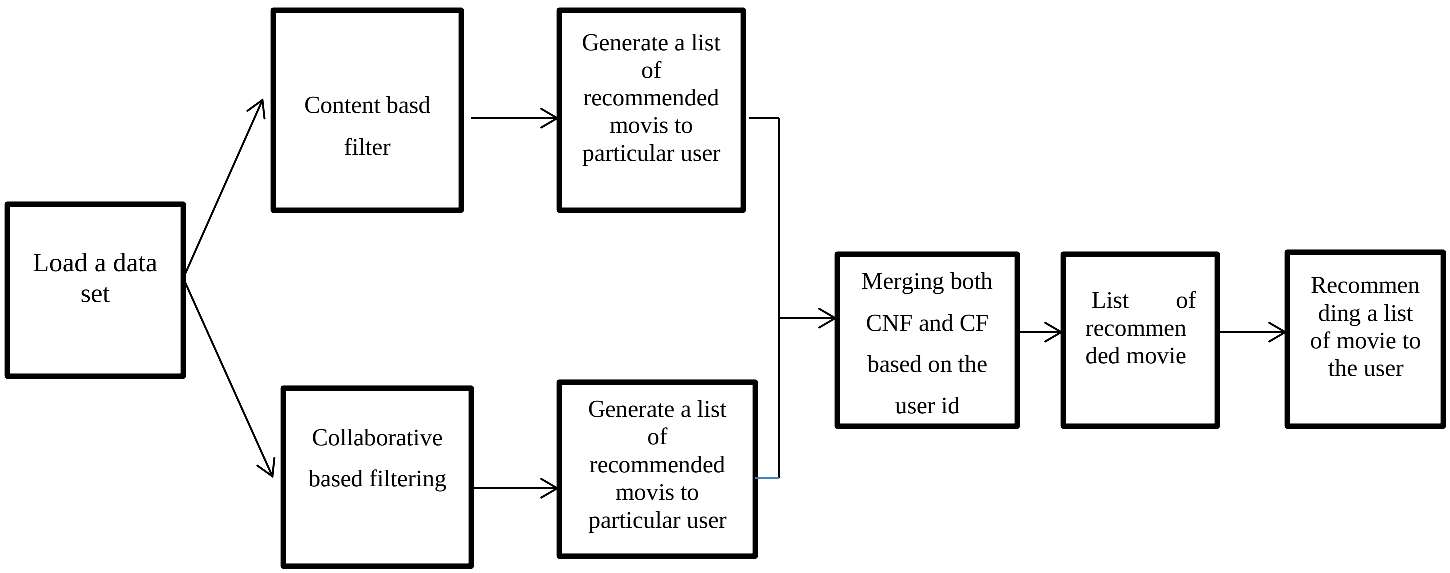


Fig:-3.3 Data Flow Diagram

Initially load the data sets that are required to build a model the data set that are required in this project are movies.csv, ratinfg.csv, users.csv all the data sets are available in the movielens dataset. Basically, two models are built in this work content based and collaborative filtering each produce a list of movies to a particular user by combining both based on the useid a single final list of movies are recommended to the particular user.

3.9. THE PHASES OF DEVELOPMENT ARE SHOWN IN FIGURE 3.3.

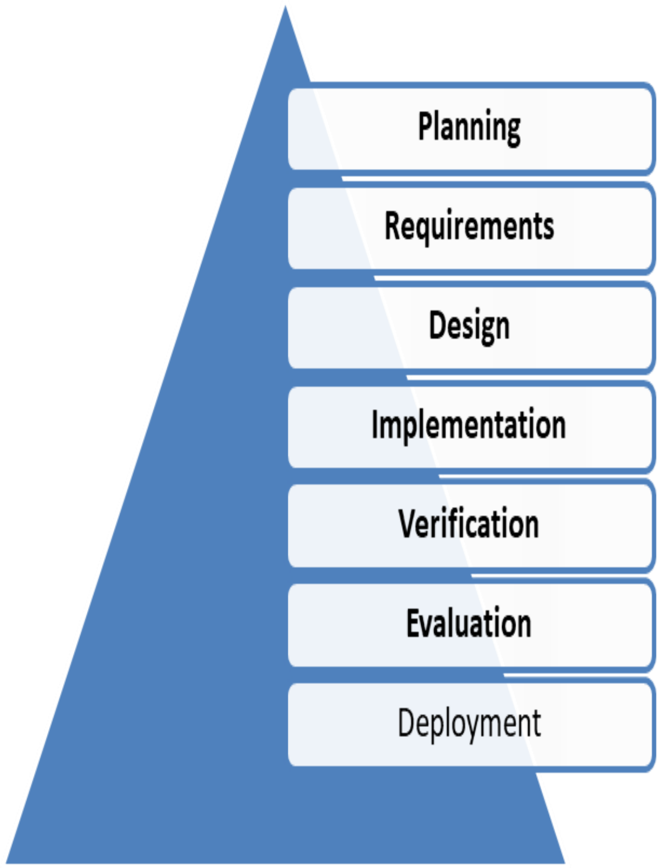


Fig 3.4: Development Phase

Planning:

As with most any development project, the first step is go through an initial planning stage to map out the specification documents, establish software or hardware requirements, and generally prepare for the upcoming stages of the cycle.

Requirements:

In this phase, requirements are gathered from customers and check by an analyst whether requirements will fulfill or not. Analyst checks that need will achieve within budget or not. After all of this, the software team skips to the next phase.

Design:

Once planning is complete, an analysis is performed to nail down the appropriate business logic, database models, and the like that will be required at this stage in the project .In the design phase, team design the software by the different diagrams like Data Flow diagram, activity diagram, class diagram, state transition diagram, etc.

Implementation:

With the planning and analysis out of the way, the actual implementation and coding process can now begin. All planning, specification, and design docs up to this point are coded and implemented into this initial iteration of the project.

Verification:

Once this current build iteration has been coded and implemented, the next step is to go through a series of testing procedures to identify and locate any potential bugs or issues that have cropped up.

Evaluation:

Once all prior stages have been completed, it is time for a thorough evaluation of development up to this stage. This allows the entire team, as well as clients or other outside parties, to examine where the project is at, where it needs to be, what can or should change, and so on

3.10. USE CASE DIAGRAMS:

Use case diagram of the recommendation system and the other subsystems are revealed in below diagram. Steps are gathered in distinct entities, the functions of which are stated in further subsections

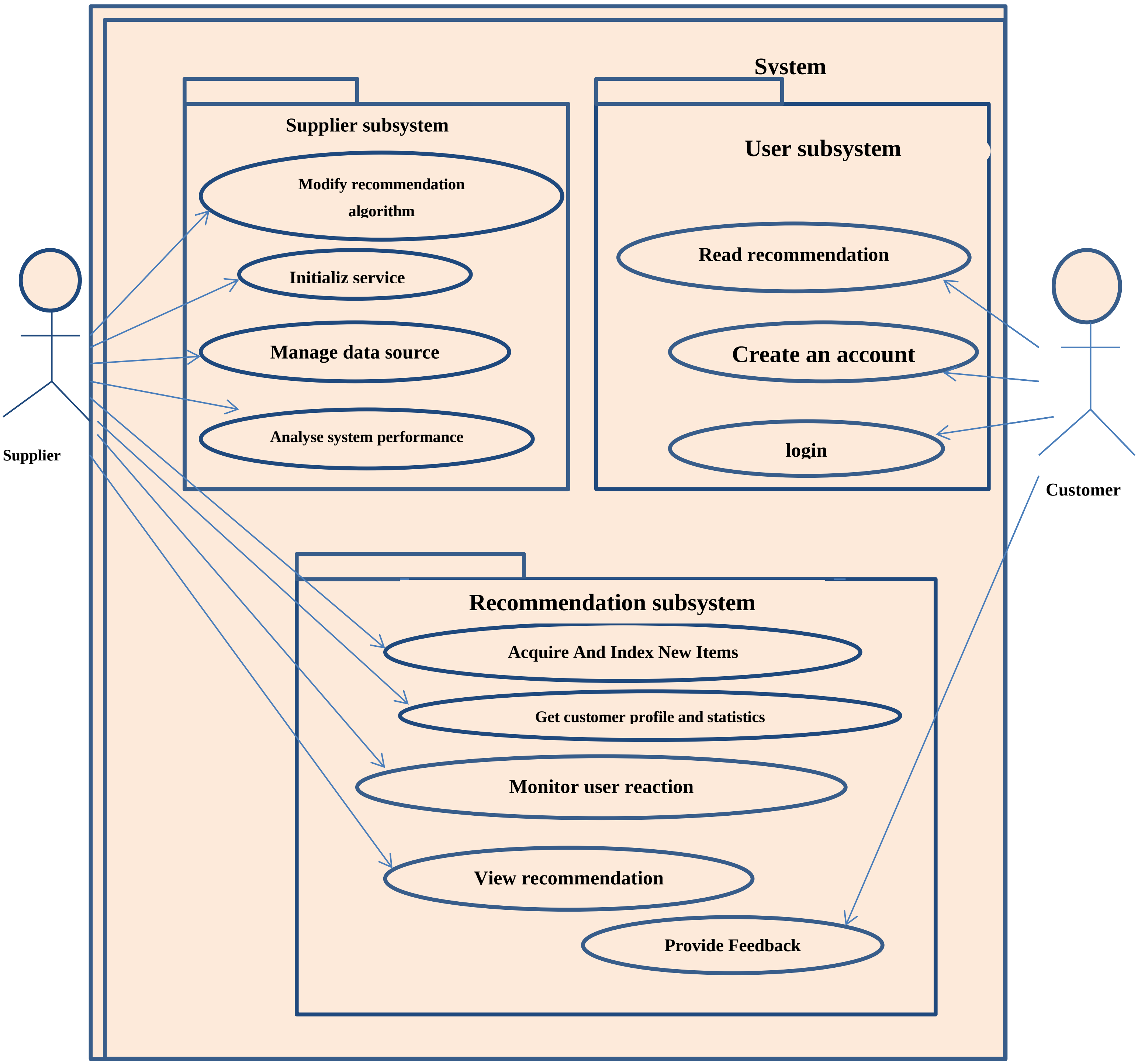
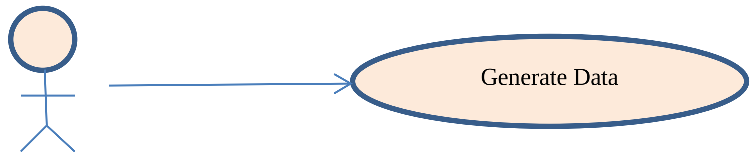


Fig :3.5: Use case diagram

USER USE CASES

Use Case: Generate Data



User

Figure 3.6: Generate Data

User can watch movie from music streaming web application. Movie information is collected according to actor information, director information, production information, artist information, time of action, user information, rating value and channel. This information will fill the database.

Every movie has unique actor, producer, director, artist, time of action, user, rating value and channel. So, this watching process will generate practical data (Figure 6).

Use Case: Get Recommendation

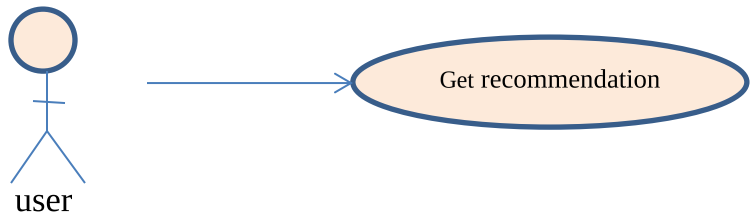
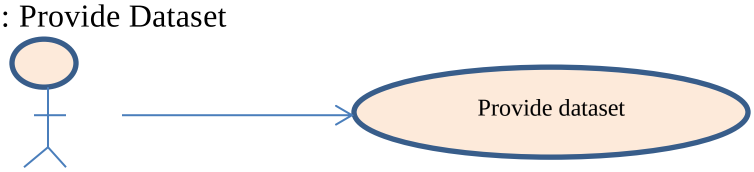


Figure 3.7: Get Recommendation

System can suggest movies as a recommendation to user based on the dataset which is refined by users' collaborative approach. The main function of our system shows these tracks based on recommendation algorithms. When a user chooses recommendation part in application, he/she will get the most important point of the project recommendation and project gives user a chance to choose track through recommended tracks according to his/her own previous choices (Figure 7).

INTERAGENT

Use Case: Provide Dataset

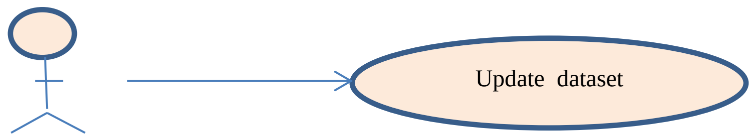


Interagent

Figure 3.8: Provide Dataset

Interagent provides dataset, in other words the big data, in cooperation with music streaming and downloading application (Figure 8).

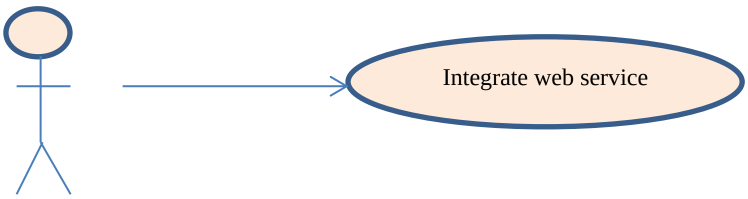
Use Case: Update Dataset



Interagent **Figure 3.9:** Update Dataset

Music streaming and downloading web application collects 1 million data every day. Interagent also delivers this 1 million data to our web service. These updates are necessary for making more accurate recommendations (Figure 9).

Use Case: Integrate Web Service



Integrate **Figure 3.10:** Integrate Web Service

After the recommendation system project is completed, inter agent integrate this web service to music streaming and downloading web application. Then, users will access to our web service and receive recommendations through the web application

CHAPTER: 4

VALIDATION OF PROPOSED WORK

4.1. VALIDATION OF PROPOSED WORK

Recommendation systems are predicting systems that radically recommend items to users or users to the items, and sometimes users to users too. Tech giants like YouTube, Amazon Prime, Netflix use similar methods to recommend video content according to their desired interest. As the internet contains huge loads of data, finding your content is very difficult and can be very time consuming, thus the recommendation plays an important role in minimizing our effort.

These systems are getting more popular nowadays in various areas such as in books, videos, music, movies, and other social network sites where the recommendation is used to filter out the information. It is a tool that is using the user’s information to improve the suggestion result and give out the most preferred choice. User/Customer satisfaction is key for building the tool. It is beneficial for both customers and companies, as the more satisfied the customer is, the more likely he/she would want to use the system for their ease, which would ultimately make revenues for the companies.

Recommendation system should always be improved as the user choice can differ from other users and if the user is not happy with the result, he/she might not use it again which is the case with our system. Our goal in this project was to keep our system very accurate compared to other recommendation techniques while making it as simple as possible. Content-based filtering has some drawbacks and a lack of accuracy and preciseness. So the proposed system is the collaborative filtering recommendation system using nearest neighbor.

4.2 Collaborative Filtering (CF)

Filters out the content according to user similar interest with other users, it basically recommends the items to users that have similar taste. It is also a popular and famous algorithm in the industries. In the memory-based techniques, there are two popular filtering algorithms. There is another technique known as model-based which is not as reliable as compared to memory-based techniques.

Recommendation systems using collaborative filtering are able to provide an accurate prediction when enough data is provided, because this technique is based on the user’s preference. User-based collaborative filtering has been very successful in the past to predict the customer’s behavior as the most important part of the recommendation system. However, their widespread use has revealed some real challenges, such as data sparsity and data scalability, with gradually increasing the number of users and items. Collaborative filtering needs a set of items that are based on the user’s historical choices. This system does not require a good amount of product features to work. An embedding or feature vector describes each item and User, and it sinks both the items and the users in a similar embedding location. It creates enclosures for items and users on its own.

Other purchaser’s reactions are taken into consideration while suggesting a specific product to the primary user. It keeps track of the behavior of all users before recommending which item is mostly liked by users. It also relates similar users by similarity in preference and behavior towards a similar product when proposing a product to the primary customer.

To improve the execution time and accuracy of the prediction problem, this chapter proposed item-based collaborative filtering applying dimension reduction in a recommendation system. It demonstrates that the proposed approach can achieve better performance and execution time for the recommendation system in terms of existing challenges, according to validation using AHP

4.3. VALIDATION STRUCTURE:

The objective of this validation study is to investigate the feasibility of the development of a passive vision based integrated moving obstacles detection and description approach that fulfills the following requirements:

- detects and classifies obstacles pertaining to a set of predefined objective
- provides depth information about each issues;
- provides information about the validation of proposed model
- is capable of determining the complete information of algorithms

4.3.1. Impacts

One of the most crucial steps in many decision-making methods is the accurate estimation of the pertinent data. This is a problem not bound in the AHP method only, but it is crucial in many other methods which need to elicit qualitative information from the decision-maker. Very often qualitative data cannot be known in terms of absolute values and unique.

Therefore, many AHP attempts to determine the relative importance, or weight, of the alternatives in terms of each criterion involved in a given decision-making problem. Pairwise comparisons are used in AHP to determine the relative importance of each alternative in terms of each criterion of django framework. In this approach the decision-maker has to express his opinion about the value of one single pairwise comparison at a time. Usually, the decision-maker has to choose his answer among various discrete choices.

To implement this validation, we first take the parameters, heading, and velocity are stored in numeric values. The prediction model provides an explicit heading estimate, while the point model provides an implicit heading based on the selection. We then take a model of the recommended movie system. We use the various domains of the dynamic paramerts to calculate. We then hypothesize that the dynamic model will continue to scenario and will likely maintain the same offset that it currently has.

We will tell researcher about the basic process involved in the recommendation system. Recommendation system works on basically on two things product details and user details. We have to collect them from the system or from the database and make decisions on the basis of ratings if a similar items were found then it will generate recommendation system otherwise, no recommendation system will be generated.

We are using AHP method for our research. In this research we are applying item based collaborative filtering. The reason behind this is because user taste may change with respect to time but item doesn’t change it remains same.

There are certain stages to make our recommendation system efficiently to respond. .

• **Data Loading:** To load the data and display accordingly we have to perform some operation like merging the two files of data set

•**Data Slicing:** Here we are removing unnecessary column and data.

• **Data Cleaning:** In the real world data if we make a table of ratings in the recommendation system ,we find that most of the user are not rating the movies and are mostly inactive. The same cases are with movies either users don’t watch or it’s get too old.

To make our computation more accurate we will remove such users and movies from our research. In our research we had used AHP method.

4.4. AHP METHODOLOGY:

The Analytic Hierarchy Process (AHP) is a multi-criteria decision-making approach and was introduced by Saaty (1977 and 1994). The AHP has attracted the interest of many researchers mainly due to the nice mathematical properties of the method and the fact that the required input data are rather easy to obtain. The AHP is a decision support tool which can be used to solve complex decision problems. It uses a multi-level hierarchical structure of objectives, criteria, sub criteria, and alternatives. The pertinent data are derived by using a set of pairwise comparisons. These comparisons are used to obtain the weights of importance of the decision criteria, and the relative performance measures of the alternatives in terms of each individual decision criterion. If the comparisons are not perfectly consistent, then it provides a mechanism for improving consistency.

One of the most crucial steps in many decision-making methods is the accurate estimation of the pertinent data. This is a problem not bound in the AHP method only, but it is crucial in many other methods which need to elicit qualitative information from the decision-maker. Very often qualitative data cannot be known in terms of absolute values. For instance,

"what is the worth of a specific computer software in terms of a user adaptivity criterion?" Although information about questions like the previous one are vital in making the correct decision, it is very difficult, if not impossible, to quantify them correctly. Therefore, many decision-making methods attempt to determine the relative importance, or weight, of the alternatives in terms of each criterion involved in a given decision-making problem. An approach based on pairwise comparisons which was proposed by Saaty (1980) has long attracted the interest of many researchers. Pairwise comparisons are used to determine the relative importance of each alternative in terms of each criterion. In this approach the decision-maker has to express his opinion about the value of one single pairwise comparison at a time. Usually, the decision-maker has to choose his answer among 10-17 discrete choices

Intensity of importance	Defination	Explanation
1	Equal importance	Two activities contribute equally to the objective
2	Weak importance of one over another	Experience and judgment slightly favor one activity over another
5	Essential or storage importance	Experience and judgment strongly favor one activity over another
7	Demonstrated importance	An activity is strongly favored and its dominance demonstrated in practice
9	Absolute importance	The evidence favoring one activity over another is of the highest possible order of affirmation
2,4,6,8	Intermediate values between the two adjacent judgments	When compromise is needed
Reciprocals of above nonzero	If activity i has one of the above nonzero numbers assigned to it when compared with activity j, then j has the reciprocal value when compared with i.	

Table 4.1: Scale of Relative Importances (according to Saaty (1980))

Source: https://bit.csc.lsu.edu/trianta/Journal_PAPERS1/AHPapls1.pdf

4.5. DEPLOYMENT:

Stage_1 Decision Metrics and Assign Weight

In table 4.5, Let C = {Cj| j = 1, 2... n} be the set of decision criteria. The data of the pair wise comparison of n sub-criteria can be summarized in an (n× n) evaluation matrix A in which every element aij (i, j = 1, 2 ... n) is the quotient of weights of the criteria. This pair wise comparison can be shown by a square and reciprocal matrix. In this matrix $a_{ij} = 1/a_{ji}$, for all experts, we would have (n× n) matrices (see table 4.5).

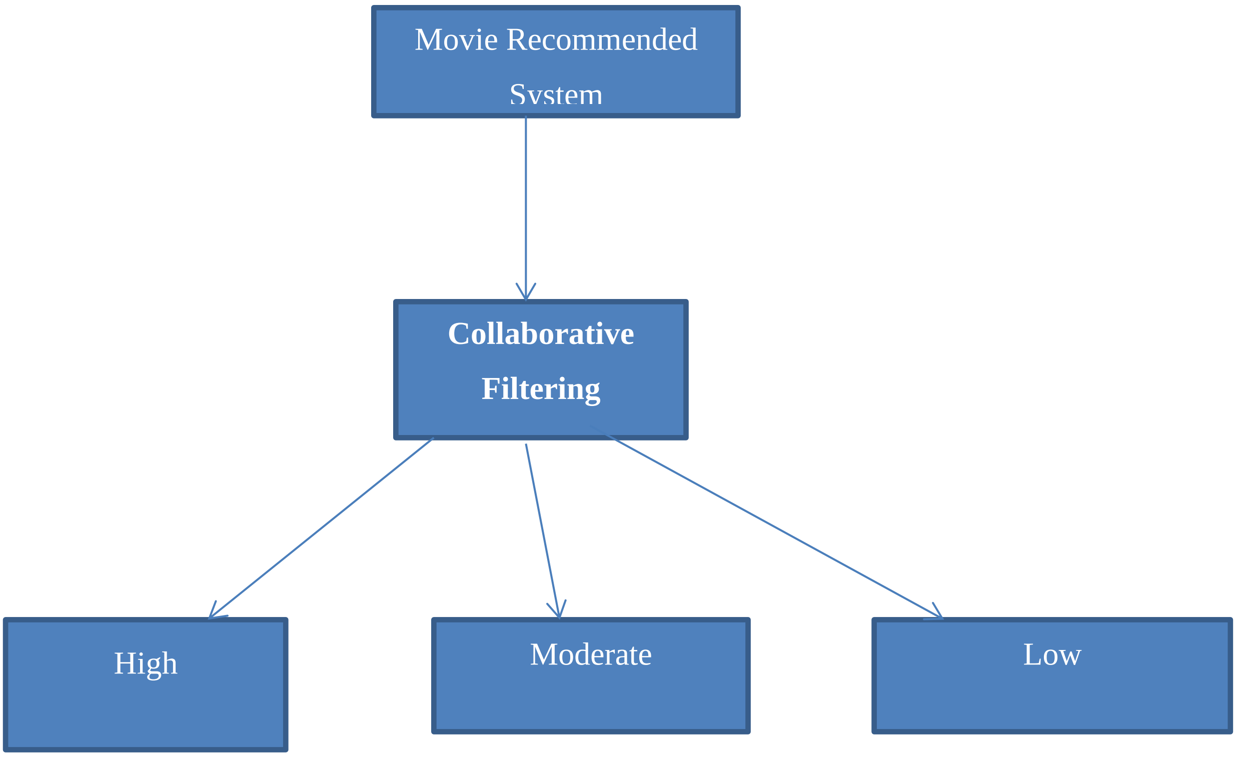


Figure 4.1: Conceptual Behaviour

Table 4.2 Assign Weight (Collaborative Filtering)			
	High	**Low**	**Moderate**
High	1	3.9	5.6
Low	0.2564	1	3.2
Moderate	0.1785	0.3125	1
	1.435	5.213	9.800

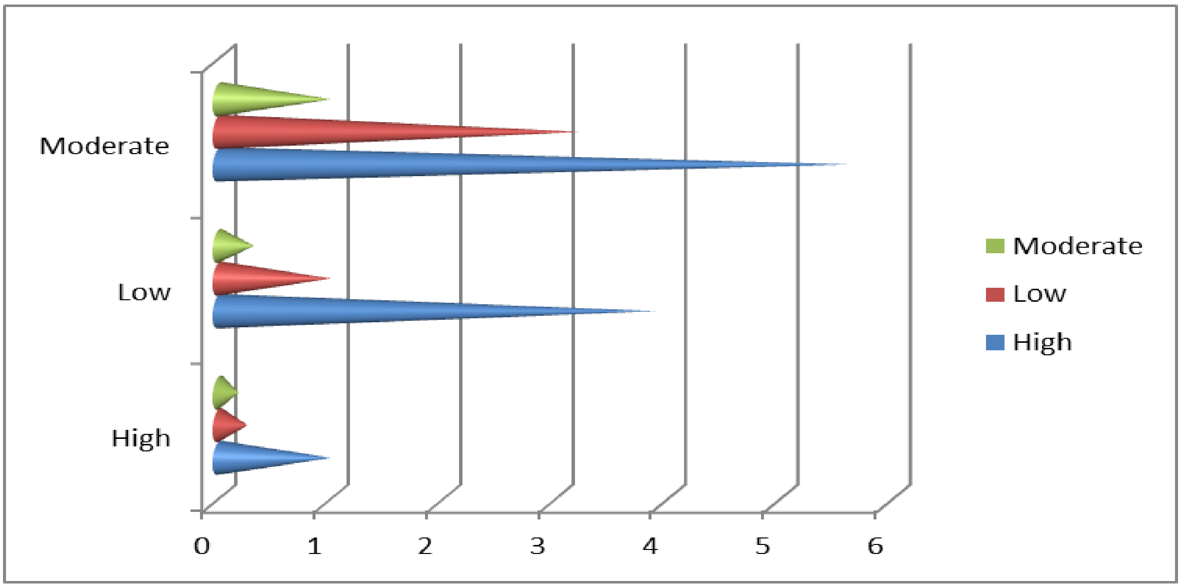


Fig 4.2: Graphical Valuation

In Table 4.3, the Accuracy resulted in the best value with an ideal priority Vector 0.6722, mainly because it was the highest evaluated in the two metrics: 0.6969 and 0.7482 (figure 4.3). Nonetheless, there are three alternatives for model that also stand out. In table 4.3, it is possible to see the average consistency index obtained from the output of the alternatives in the test phase.

As a result of the curse of dimensionality, it is possible to use the AHP to calculate the options among different models and justify the model's accuracy. The new approach has been introduced to solve the most important alternatives, and the details of this approach are provided in Table 4.3 with consistency index for verifying the stage calculations.

Table 4.3 Normalized Metrics			
	High	**Low**	**Moderate**
High	0.6969	0.7482	0.5714
Moderate	0.1787	0.1918	0.3265
Low	0.1244	0.0600	0.1020
Eigen Vector		**Priority**	
0.9645		0.6722	
1.2111		0.2324	
0.9357		0.0955	
Eigen Value 3.114		0.0557	

In table 4.4, we have calculated to overall priority of each criteria respect to model weight. We are observed that high values of very effective in this research work. Table 4.5 are given the finalize metrics in the form of priority High Low Moderate context. Accuracy value is the maximum effective constraints which provides verifying and validated of research.

Table 4. 4 Calculate overall priority			
	High	**Low**	**Moderate**
High	0.6722	0.653	0.600
Moderate	0.2324	0.251	0.200
Low	0.0955	0.096	0.200

Table 4.5 Finalize Metrics	
High	**0.6252**
Moderate	0.2137
Low	0.1611
Highest Priority = Highest Score	

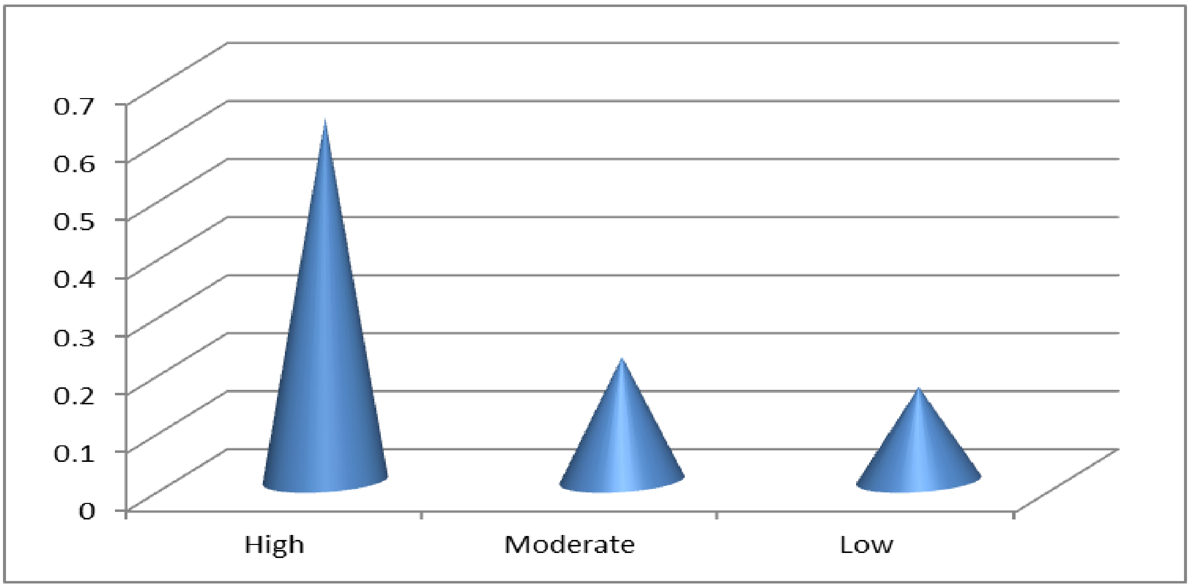


Fig 4.3: Final Structure

In this research, three models are implemented for correcting the inconsistency of the AHP pairwise comparison matrix; these methods are based on High, Low and Moderate. Firstly, simulations are performed with training, validation, and testing to compare both methods. The Collaborative Filtering method has a behavior similar to High value in CR reduction but with a better accuracy rate in predicting previously unknown inputs presented to the network and provides the advantage of a significantly of Collaborative Filtering method. In fig 4.3, it is possible to see the final weight obtained from the AHP evaluations. It is compared to the three models of the original input elements; we can say that Collaborative Filtering method has managed to play a positive role between them. Some Observations and keynotes have given in below:

- An evaluation study of the Analytic Hierarchy Process (AHP) and MLP approach is also made while extracting the weights of criteria for Models and their criteria and alternatives.

- In this evaluation, we have first done to predict the disease through MLP algorithms with specific accuracy and prevention of selection at an early stage. In the second stage, we accepted the AHP process to illustrate the model dependency variable.
- In the Clarity concept, we have done the work in one segment, such as High, Low and Moderate based on the p value of table 4.2.
- Table 4.2 illustrates the weight according to the model, and we found that High, Low and Moderate is gained heights weight and supports table 4.5 results.

CHAPTER 5

RESULT AND COMPARATIVE STUDY

5.1. ENVIRONMENT

Since our project is movie recommendation system .one can develop a movie recommendation system by using either content based or collaborative filtering or combining both. In our project we have developed a hybrid approach i.e combination of both content and collaborative filtering .Both the approaches have advantages and dis-advantages

- **In content based filtering** the it based on the user ratings or user likes only such kind of movie will recommended to the user.
    - Advantages: it is easy to design and it takes less time to compute
    - Dis-advantages: the model can only make recommendations based on existing interests of the user. In other words, the model has limited ability to expand on the users' existing interests.
- **In Collaborative filtering** the recommendation is comparison of similar users.
    - Advantages: No need domain knowledge because the embeddings are automatically learned. The model can help users discover new interests. In isolation, the ML system may not know the user is interested in a given item, but the model might still recommend it because similar users are interested in that item.
    - Dis-advantages: The prediction of the model for a given (user, item) pair is the dot product of the corresponding embeddings. So, if an item is not seen during training, the system can't create an embedding for it and can't query the model with this item. This issue is often called the cold-start problem. The hybrid approach will resolves all these limitations by combining both content and collaborative filtering

5.2. COMPARATIVE ANALYSIS

After we have obtained the findings, it is strongly advised that you compare the work that was proposed with other recent work that has been done in the same area. Manoj et al. (2015) have implemented a method for a movie recommendation system that is based on the weighting of criteria, and this method is quite similar to the one that we have suggested here. In the future, we want to concentrate on improving its user interface as well as its weaknesses.

	Weight of Proposed work	Ranking of Proposed work	Weight of Existing work	Ranking of Existing work
Aggregate	**0.6252**	1	0.60	1
	0.2137	2	0.54	2
	0.1611	3	0.48	3
Moderate	0.653	1	0.42	3
	0.251	2	0.36	2
	0.096	3	0.30	1
High	0.6722	1	0.61	3
	0.2324	2	0.48	2
	0.0955	3	0.42	1
Low	0.600	1	0.45	3
	0.200	2	0.36	2
	0.200	3	0.06	1

Table 5.1 Comparison of ranks between proposed work and existing work (Manoj et.al(2015)

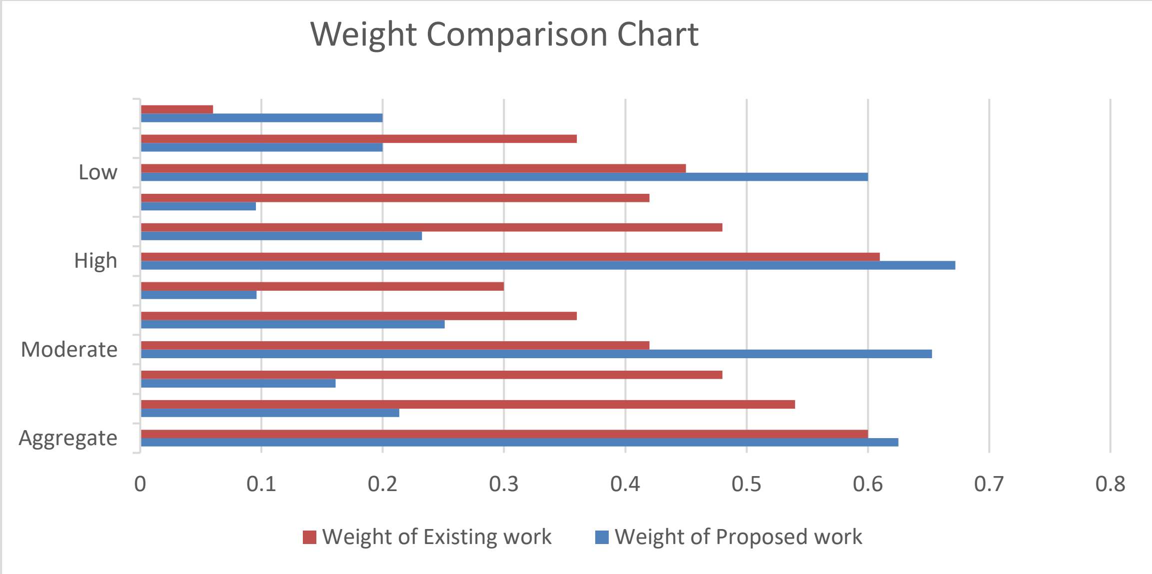


Figure 5.1: Weight Comparison Chart

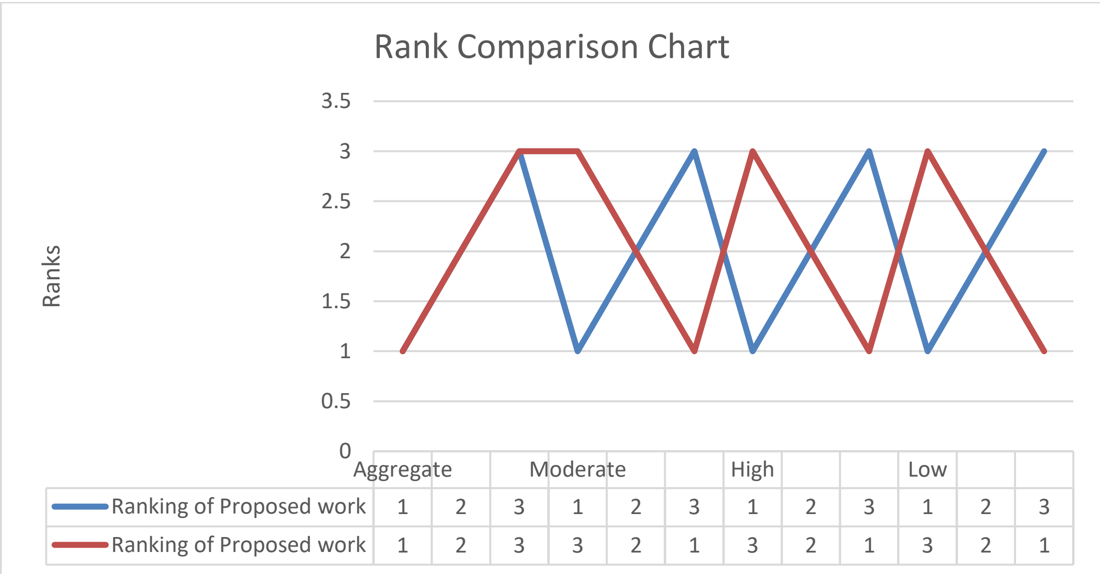


Figure 5.2: Rank Comparison Chart

5.3. SNAPSHOT OF IMPLEMENTATION:

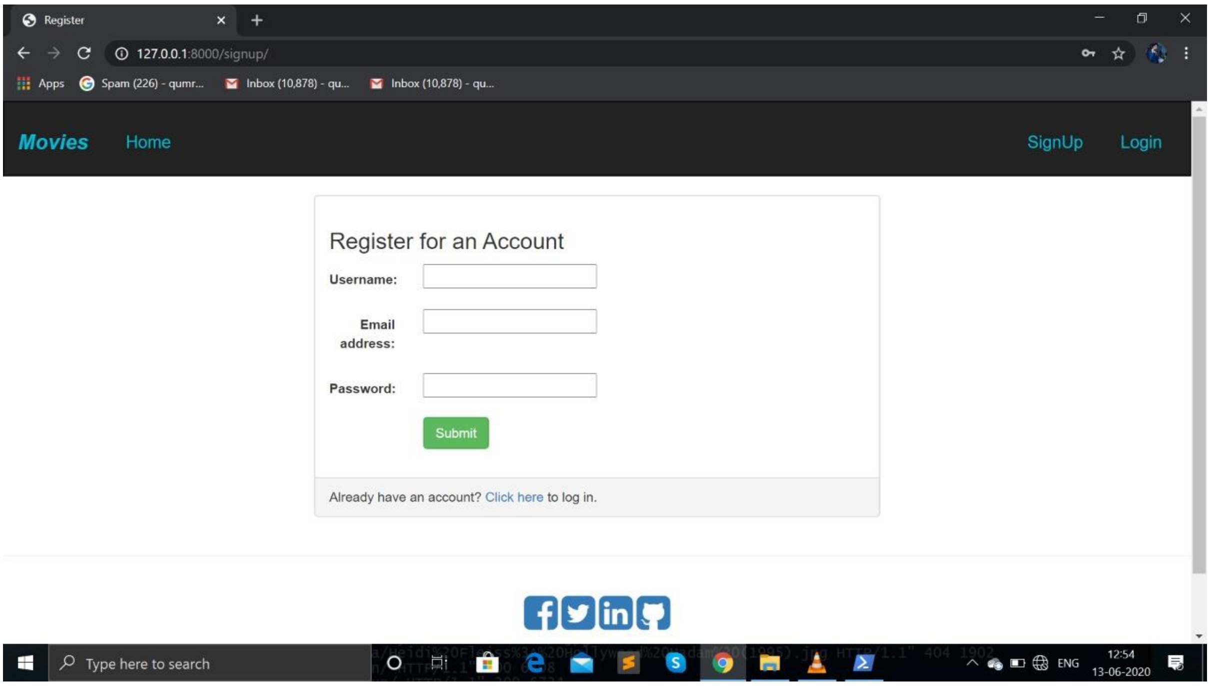


Fig. 5.3: Registering a user

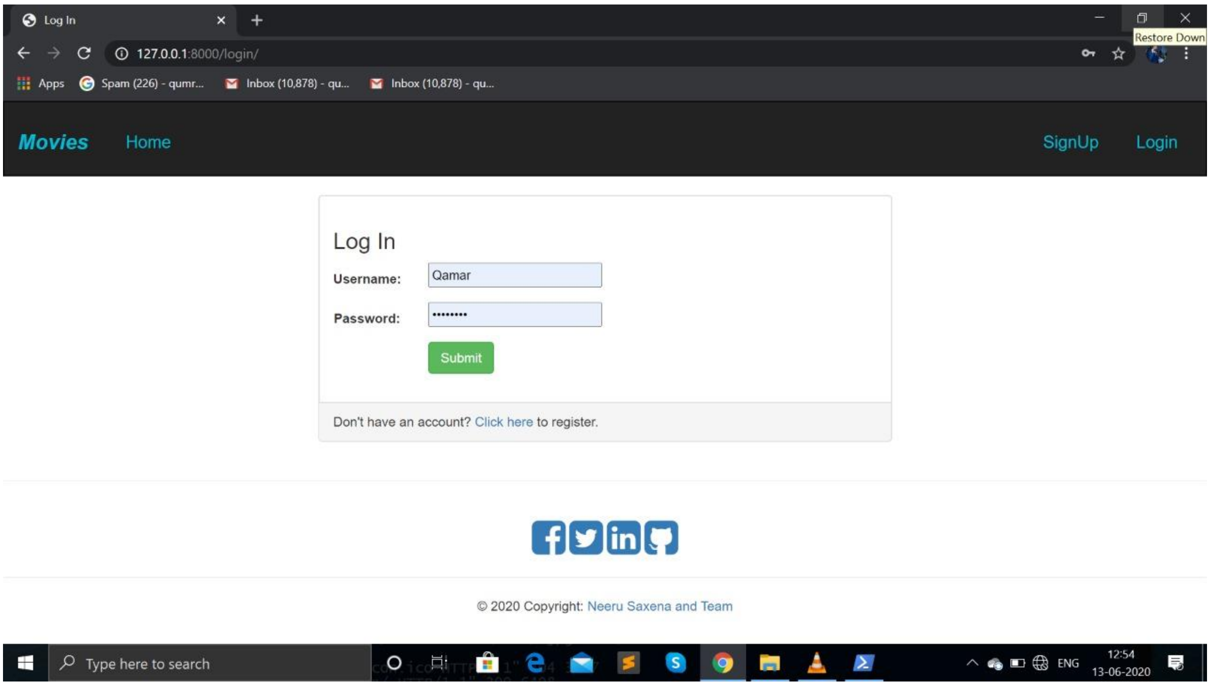


Fig. 5.4: Login a user interface

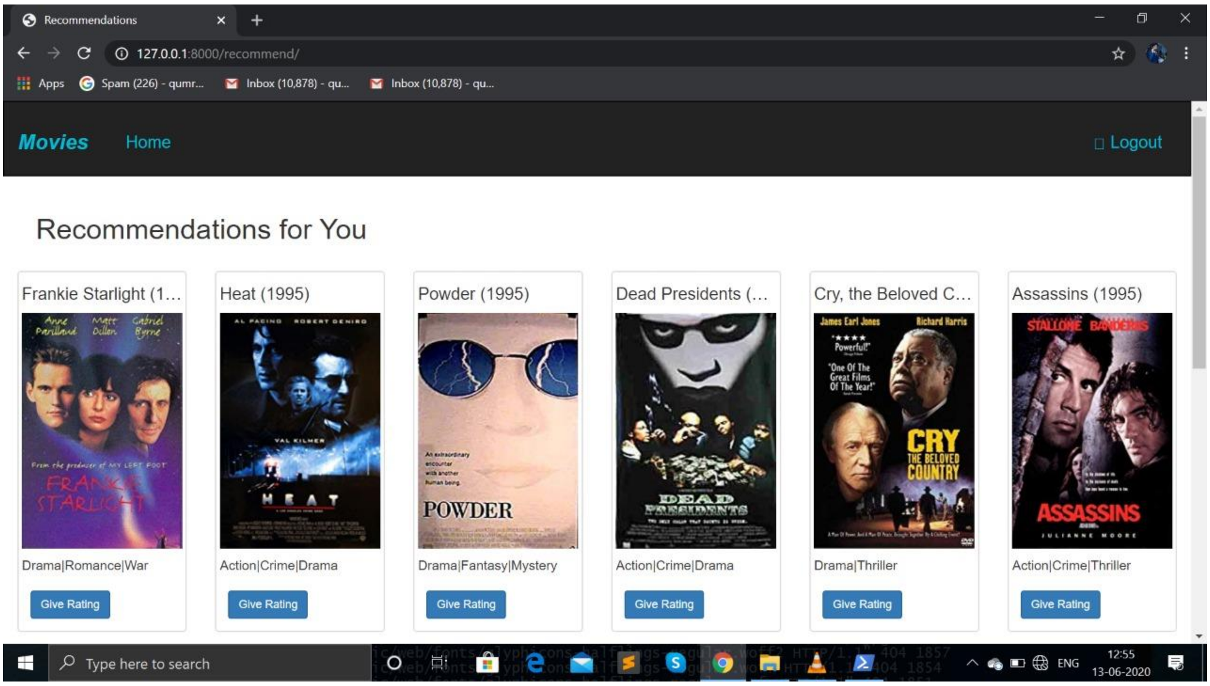


Fig. 5.5: Get Recommendation Interface

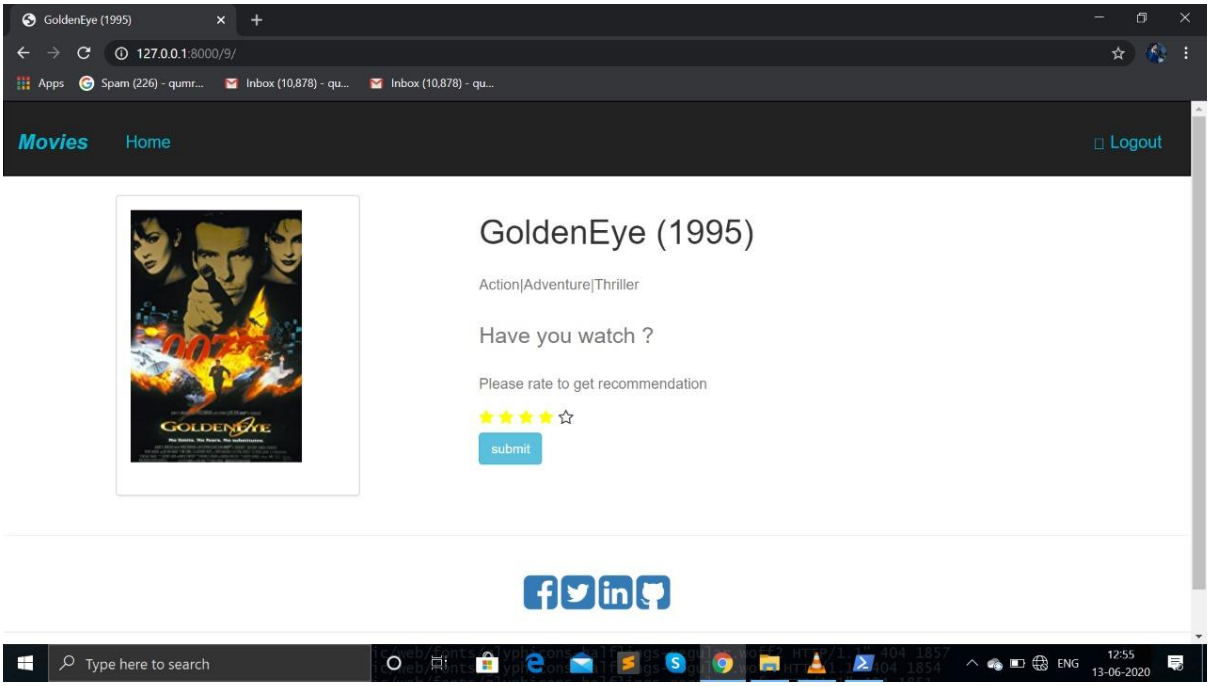


Fig. 5.6: Rate a Movie Interface

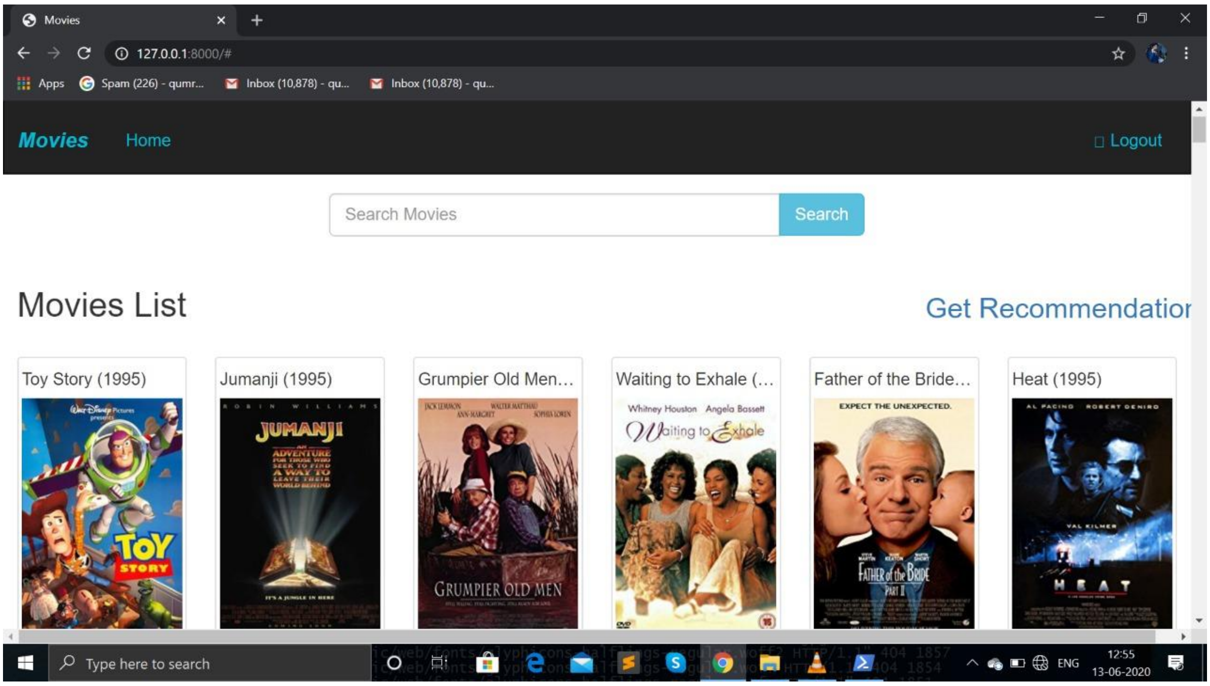


Fig. 5.7: Getting similar Users

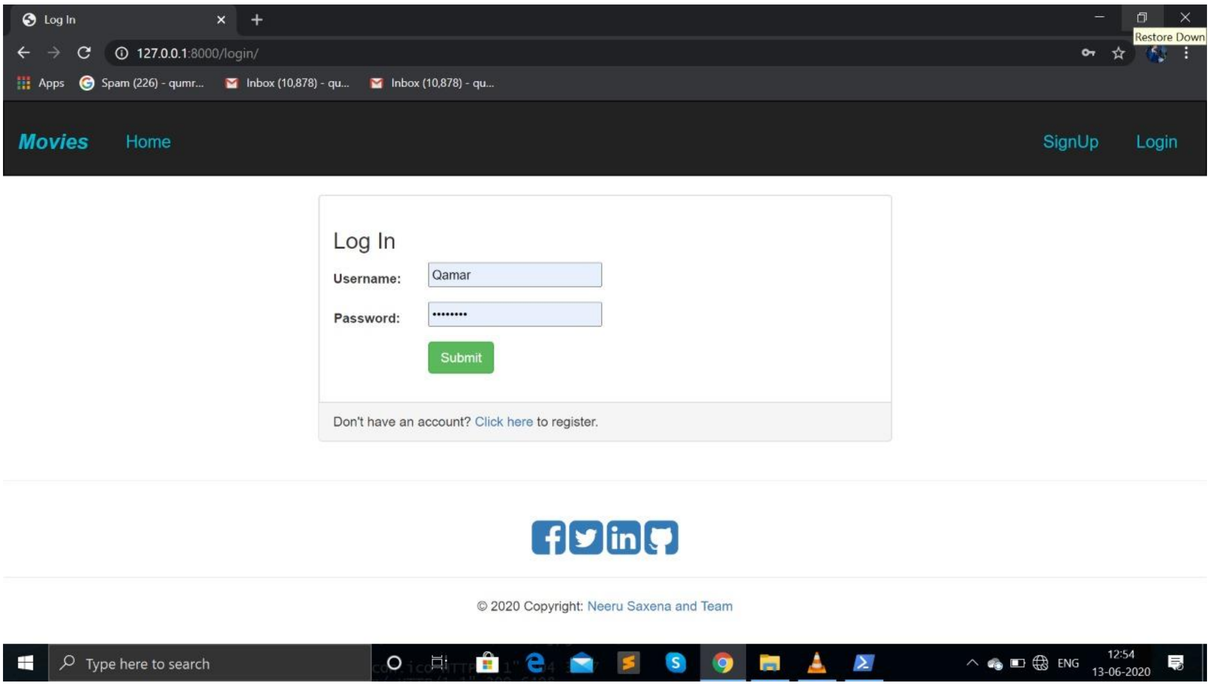


Fig. 5.8: Logging out a User

CHAPTER: 6

CONCLUSION AND FUTURE SCOPE

6.1 CONCLUSION

There are new routes for obtaining personal information online thanks to the advised solutions. A common problem with information retrieval systems is an overabundance of information, and this solution helps to relieve that problem while simultaneously providing customers with quick, convenient access to products and services that may otherwise be unavailable. Based on the input that customers have given us in the past and our strategy to respond to their unique needs and interests, we recommend movies to them. As a result of this method, known as AHP, the recommendations model becomes more accurate. In addition, the system is more responsive and the data collecting procedure is more accurate as a result of this. In addition to confirming the model, this approach also improve the system's performance. When it comes to discovering new things, the Recommendation system may be a lifesaver. Certain restrictions prevent these systems from suggesting effectively to their consumers. Even though Collaborative Filtering is the most effective and powerful method, this algorithm has a high runtime and confronts challenges like as data sparsity, which may be eliminated by employing a Hybrid movie recommendation system. However, there are advantages and disadvantages to each option. In our suggested validation method, we can deal with quite a large quantity of data with ease. We plan to address its weaknesses and improve its user interface in the future

In our project, we have developed a Movie Recommender System using Collaborative Filtering Recommendation System. In this system, we used python for the development of website, which provides high level of abstraction and readability and clean code to our project. To store the information of the user’s relationship, labels, data and properties, we used Relational Database, i.e., SQLite. Apart from SQLite, in our work, we have used Scipy, a Python library, which provides the easy use of mathematical formula and equations for our application. In our movie recommendation website, the first step is to register the users before using our movie recommendation website. After registration, users can get recommendation the important information with the other users, who are connected with the website.

6.2 FUTURE SCOPE

In future, we would like to work on the following issues:

1. To search the movie based on the requirements of the users
2. To recommend the movie to the interested users more accurately
3. To a develop the RS under dynamic environment.
4. 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.
5. 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. Ahuja R, Solanki A, Nayyar A (2019) Movie recommender system using k-means clustering and k-nearest neighbor. Proc 9th IntConf Cloud Comput Data SciEngConflu 2019 263–268.
2. Indira K, Kavithadevi MK (2019) Efficient machine learning model for movie recommender.pdf
3. Yang D, Zhou Z (2013) Personalized mining of preferred paths based on web log. Proc 2013 IEEE 11th IntConf Electron Meas Instruments, ICEMI 2013 2:993–997.
4. Wu CSM, Garg D, Bhandary U (2019) Movie recommendation system using collaborative filtering. Proc IEEE IntConfSoftwEngServSci ICSESS 2018-November 11–15. https://doi.org/10.1109/ICSESS.2018.8663822
5. Xu X, Zhang Y (2018) Collaborative filtering recommendation algorithm based on hybrid similarity. 2017 IntConfComputSyst Electron Control ICCSEC 2017 2018:1372–1375. https://doi.org/10.1109/ICCSEC.2017.8447000
6. Yang D, Zhou Z (2013) Personalized mining of preferred paths based on web log. Proc 2013 IEEE 11th IntConf Electron Meas Instruments, ICEMI 2013 2:993–997.
7. M. Shah, D. Parikh and B. Deshpande, "Movie Recommendation System Employing Latent Graph Features in Extremely Randomized Trees," in Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies (ICTCS '16), New York, 2016.
8. Y. Deldjoo, M. Elahi, M. Quadrana and P. Cremonesi, "Using visual features based on MPEG-7 and deep learning for movie recommendation," Int J Multimed Info Retr, vol. 7, p. 207–219, 2018.
9. J. K. Leung, I. Griva and W. G. Kennedy, "Making Use of Affective Features from Media Content Metadata for Better Movie Recommendation Making," arXiv, 2020.

10. S. Reddy, S. Nalluri, S. Kunisetti, S. Ashok and B. Venkatesh, "Content-Based Movie Recommendation System Using Genre Correlation," in Smart Intelligent Computing and Applications, Singapore, 2019.
11. H. Li, J. Cui, B. Shen and J. Ma, "An intelligent movie recommendation system through group-level sentiment analysis in microblogs," Neurocomputing, vol. 210, pp. 164-173, 2016.
12. P. P. Adikara et al. ISSN 2502-3357 (online) | ISSN 2503-0477 (print) regist. j. ilm. teknol. sist. inf. 7 (1) January 2021 31-42 Movie recommender systems using hybrid model based on graphs with co-rated … http://doi.org/10.26594/register.v7i1.2081 \
13. O.-J. Lee and J. J. Jung, "Explainable Movie Recommendation Systems by using Story-based Similarity," in Explainable Smart Systems 2018 (ExSS ’18), Tokyo, 2018.
14. J. Li, W. Xu, W. Wan and J. Sun, "Movie recommendation based on bridging movie feature and user interest," Journal of Computational Science, vol. 26, pp. 128-134, 2018.
15. W. E. Winkler, "String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage," The Educational Resources Information Center (ERIC), Washington, DC, 1990.
16. C. D. Manning, P. Raghavan and H. Schütze, An Introduction to Information Retrieval, Cambridge: Cambridge University Press, 2008.
17. A. Vukotic, N. Watt, T. Abedrabbo, D. Fox and J. Partner, Neo4j in Action, Greenwich, CT, United States: Manning Publications Co, 2014.
18. Choi, Sang-Min, Sang-Ki Ko, and Yo-Sub Han. "A movie recommendation algorithm based on genre correlations." Expert Systems with Applications 39.9 (2012): 8079-8085.

19. Lekakos, George, and Petros Caravelas. "A hybrid approach for movie recommendation." Multimedia tools and applications 36.1 (2008): 55-70.
20. Das, Debashis, Laxman Sahoo, and Sujoy Datta. "A survey on recommendation system." International Journal of Computer Applications 160.7 (2017).
21. Dietmar Jannach, Gerhard Friedrich; “Tutorial: Recommender Systems”, International Joint Conference on Artificial Intelligence, Beijing, August 4, 2013.
22. Gaurangi, Eyrun, Nan; “Movie GEN: A Movie Recommendation System”, UCSB.
23. Harpreet Kaur Virk, Er. Maninder Singh,” Analysis and Design of Hybrid Online Movie Recommender System “International Journal of Innovations in Engineering and Technology (IJIET)Volume 5 Issue 2,April 2015.
24. Manoj Kumar, D.KYadav, Ankur Singh, Vijay Kr. Gupta,” A Movie Recommender System: MOVREC” International Journal of Computer Applications (0975 –8887) Volume 124 – No.3, August 2015.
25. Prerana Khurana , Shabnam Parveen; ‘Approaches of Recommender System: A Survey’; International Journal of Computer Trends and Technology (IJCTT) Volume 34 Number 3 - April 2016.
26. Manoj K, el.al., “A Movie Recommender System: MOVREC”,International Journal of Computer Applications (0975 – 8887) Volume 124 – No.3, August 2015 7
27. J. Bobadilla, F. Ortega, A. Hernando, A. Gutierrez Recommendation system survey

PLAGIARISM CHECK REPORT

ABSTRACT

Plagiarism Scan Report

Summary	
Report Generated Date	7 July, 2022
Plagiarism Status	100% Unique
Total Words	167
Total Characters	1022
Any Ignore URL Used	

Content checked for Plagiarism:

Abstract…….

Report generated by: smallseotools.com

CHAPTER 1

Plagiarism Scan Report

Summary	
Report Generated Date	7 July, 2022
Plagiarism Status	100% Unique
Total Words	764
Total Characters	5,180
Any Ignore URL Used	

Content checked for Plagiarism:

CHAPTER: 1

INTRODUCTION……

Introduction of Information Filtering System …….

Classification of Information Filtering System …

Types Of Recommendation System …

Report generated by: smallseotools.com

Plagiarism Scan Report

Summary	
Report Generated Date	7 July, 2022
Plagiarism Status	100% Unique
Total Words	916
Total Characters	5,842
Any Ignore URL Used	

Content checked for Plagiarism:

Motivation…

Background …

Scope of the project….

Methodology………

Dissertation Outline…..

Report generated by: smallseotools.com

CHAPTER 2

Plagiarism Scan Report

Summary	
Report Generated Date	7 July, 2022
Plagiarism Status	100% Unique
Total Words	668
Total Characters	4185
Any Ignore URL Used	

Content checked for Plagiarism:

Chapter 2

Literature Review…..

Report generated by: smallseotools.com

Plagiarism Scan Report

Summary	
Report Generated Date	7 July, 2022
Plagiarism Status	100% Unique
Total Words	751
Total Characters	4,808
Any Ignore URL Used	

Content checked for Plagiarism:

Literature review

Report generated by: smallseotools.com

Plagiarism Scan Report

Summary	
Report Generated Date	7 July, 2022
Plagiarism Status	100% Unique
Total Words	1,172
Total Characters	7,386
Any Ignore URL Used	

Content checked for Plagiarism:

Literature review …….

Report generated by: smallseotools.com

CHAPTER 3

Plagiarism Scan Report

Summary	
Report Generated Date	7 July, 2022
Plagiarism Status	100% Unique
Total Words	763
Total Characters	5,008
Any Ignore URL Used	

Content checked for Plagiarism:

Chapter 3

Proposed Methodology…

Software And Hardware Requirment……

Objectives …….

Problem Statement….

Report generated by: smallseotools.com

Plagiarism Scan Report

Summary	
Report Generated Date	7 July, 2022
Plagiarism Status	100% Unique
Total Words	660
Total Characters	4,022
Any Ignore URL Used	

Content checked for Plagiarism:

Technology Used In Our Project Work…

Algorithm Used …….

System Architecture Of Proposed System…..

Report generated by: smallseotools.com

Plagiarism Scan Report

Summary	
Report Generated Date	7 July, 2022
Plagiarism Status	100% Unique
Total Words	679
Total Characters	4,560
Any Ignore URL Used	

Content checked for Plagiarism:

Dataflow …….

Use case diagram

Report generated by: smallseotools.com

CHAPTER-4

Plagiarism Scan Report

Summary	
Report Generated Date	7 July, 2022
Plagiarism Status	100% Unique
Total Words	569
Total Characters	3,642
Any Ignore URL Used	

Content checked for Plagiarism:

Chapter-4

Validation and verification of proposed work

Study impact….

Collaborative filtering…….

Report generated by: smallseotools.com

Plagiarism Scan Report

Summary	
Report Generated Date	7 July, 2022
Plagiarism Status	100% Unique
Total Words	848
Total Characters	5,416
Any Ignore URL Used	

Content checked for Plagiarism:

Validation structure……

AHP Methodology…..

Report generated by: smallseotools.com

Plagiarism Scan Report

Summary	
Report Generated Date	7 July, 2022
Plagiarism Status	100% Unique
Total Words	634
Total Characters	3,765
Any Ignore URL Used	

Content checked for Plagiarism:

Deployment……

Report generated by: smallseotools.com

CHAPTER-5

Plagiarism Scan Report

Summary	
Report Generated Date	7 July, 2022
Plagiarism Status	100% Unique
Total Words	478
Total Characters	2,804
Any Ignore URL Used	

Content checked for Plagiarism:

Chapter-5

Environment……..

Result And Comparative Study

Report generated by: smallseotools.com

CHAPTER-6

Plagiarism Scan Report

Summary	
sReport Generated Date	7 July, 2022
Plagiarism Status	100% Unique
Total Words	493
Total Characters	3,091
Any Ignore URL Used	

Content checked for Plagiarism:

Chapter-6

Conclusion and Future Scope

Conclusion……..

Future scope…….

Report generated by: smallseotools.com

PUBLICATION FROM THIS WORK

1. *My first paper "* ***A Review Study on Movie Recommendation System****" was published in* International Journal Of Creative Research Thoughts, Volume 10 ~ Issue 04 (April – 2022). You can open it by clicking on this link. https://ijcrt.org/papers/IJCRT2204297.pdf html

2. My second paper "**An Analytical study on Recommendation systems Using Collaborative Filtering: AHP Perspective**" was published in International Journal of Creative Research Thoughts.

IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)
An International Open Access, Peer-reviewed, Refereed Journal

A REVIEW STUDY ON MOVIE RECOMMENDATION SYSTEM

Ismat Anjum [1], Anwar Ahmad Shaikh [2]

[1]CSE Integral University, Lucknow, India
[2]Department of CSE Integral University, Lucknow

Abstract: A movie recommendation is essential in our social life since it has the ability to provide more enjoyment than other forms of entertainment. Depending on the users' interests or the popularity of the films, a system like this may provide them with a selection of movies to watch. A recommendation system is used for the purpose of suggesting products to purchase or to view. In the meanwhile, consumers cannot enjoy all accessible new releases or unseen movies owing to their restricted time. They also still need to pick which movies to view when they have extra time. This scenario is not favourable for the movie sector too. In order to satisfy consumers in picking what movies to watch and to improve movie sales, a system that can recommend relevant movies is necessary, either unseen in the past or recent releases. This study focuses on the review on hybrid technique, a blend of content-based and collaborative filtering, utilising a new perspective.

***Index Terms* -.** Movie recommendation, Filtering method, Hybrid Method

I Introduction

The recommendation system is a component of everyday life where individuals rely on knowledge to make decisions about what they want to do [14]. Collaboration filtering models take into account a user's prior purchases, as well as the judgments made by other users who have made comparable purchases or given numerical ratings to the things they purchased. After that, several models are employed to predict what the user would be interested in (or how they rate certain goods). However, despite the fact that several approaches have been established in the past. Although search is still used in many apps, which customize recommendations and cope with a lack of accuracy, it is still being utilised because of its widespread use. These demands pose a few difficulties. Alternating Least Squares, Singular Value decomposition, K-Nearest Neighbor method, and Normal predictor algorithm have been utilised by various academics to address this problem. Memory-based and model-based collaborative filtering approaches are the two main types. Methods relying on memory may be simply adapted to use all the ratings before the filtering phase, thereby ensuring that their findings are always up to date. On the other hand, a model-based system such as a neural network, develops a model that learns from the knowledge of user-item evaluations and recommends new goods. In order to produce a stronger and more accurate recommendation system, the recommender system still has to be improved. As a result of the system's recommendations, customers may learn more about products that may be of interest to them. In this study, a variety of approaches are discussed. The needs of life are never enough to satisfy a person's self-satisfaction, and so is the constant need for enjoyment in daily life. Watching movies is one of the fun things to do in your spare time. Movies are universally popular, regardless of the genre or the age of moviegoers. This is why the movie industry is so lucrative [11]. Many films or movies are released at the same time in order to satisfy the audience and make money. However, some people, because to time or money constraints, are unable to see all of the new releases. Some people prefer to view movies at a later time, and this might lead to them forgetting what they were supposed to see. To jog their memory about what they wish to see, most consumers turn to the Internet, such as online retailers selling or renting movies [10]. Streaming video-on-demand services are now readily available on the web and on smart phones, thanks to the use of certain video-streaming applications. Smart televisions and set-top boxes with video-streaming capabilities are becoming more commonplace nowadays. Categorization methods that employ a variety of data organization and classification methodologies are common in the field of machine learning. Data for training classifiers is possible [8].

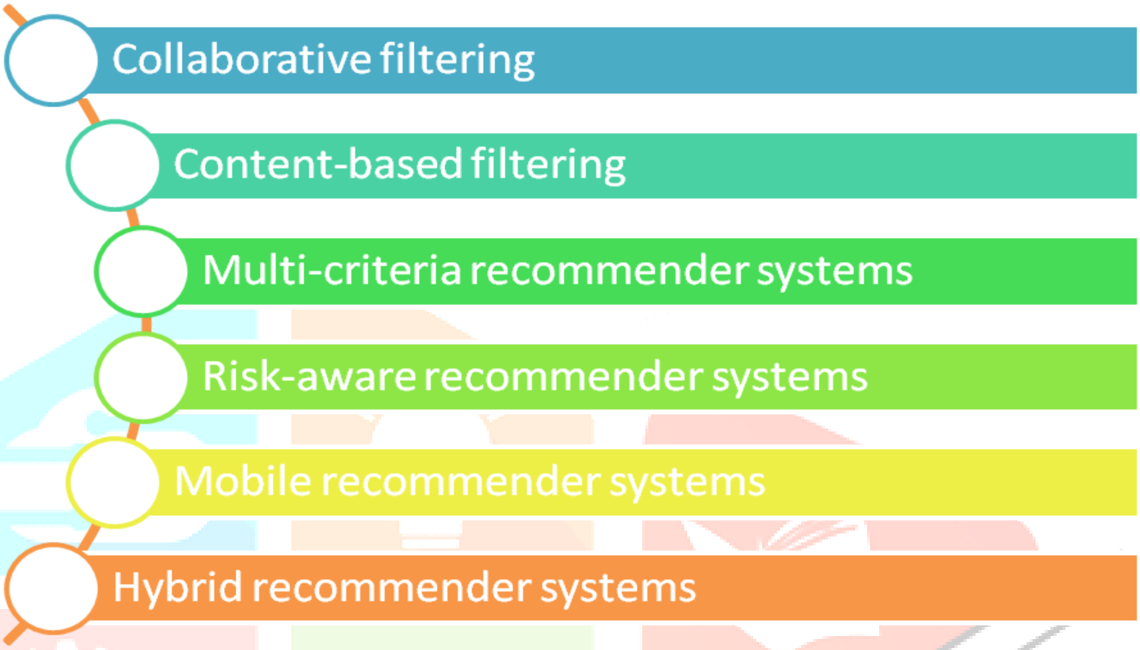


Fig 1: Method for recommendations

II Review Literature

In a work by Ahuja et al. (2019), a recommendation strategy that utilises both KNN algorithms and the K-means technique is envisioned. The client is approached in order to obtain information about the finer points. The user's userid, gender, and age are all provided by the user. The pandas module divides the data generally according to the customer and movies into separate dfs in the processing module. For the K-means module, the movie genre can be shown on an edge of data. WCSS determines the appropriate number of clusters. Pearson's correlation similarity and regularization model uses a matrix to calculate the connection. When determining film ratings, the algorithm employs KNN predictions and the UC grid to compare results. A pre-processing step eliminates outliers in both Indira and Kavithadevi (2019) and the present study (NPCA-HAC). This is followed by the use of feature selection and principal component analysis. K-means and HAC are used to group the selected characteristics. A trust rating algorithm is used to rate the clustered groupings. The clustering approach utilised in this study resulted in a loss of data owing to dimensionality reduction. Prediction performance and scalability are mutually exclusive. As a result of collaborative filtering, data sparsity, excessive computing complexity, and over-specification can be reduced. Combination models are suggested to provide a real-time item that is tailored to the needs of the consumers. Final recommendation list categorization is based on the MP neuron model. Scalability is an issue that has not been addressed in the suggested paradigm. The new item-centered strategy employs CF and CBF techniques and proposes items based on feelings. Reviews and comments on a certain product are used to extract feelings. Emotions can be used to produce item-to-item similarities. It's a good paradigm, however it doesn't take into consideration scalability and computing time. The method of discovering and crafting a film by taking into account the cinema formats of potential audiences. Users are grouped together based on their shared tastes and the ratings they have given to films they have seen. RNN may be used to evaluate and create movies, as well as to discover patterns in the viewing habits of similar groups of users. Three methods are employed in [3] and in this paper: a basic RS, a content-based approach, and a CF approach. Machine learning is employed in this project. The chart for the basic recommender system is made using IMDB's method for weighted rating. Two further techniques are followed. Sparsity, new user problems, and decreasing computing efficiency all contribute to decreased performance. It has been shown that item-based collaborative filtering (ICF) is superior to user-based CF in terms of analysis and data processing complexity, as demonstrated in this work. Working performance may be improved by utilising item content and feature vectors. A sign-up system collects the user's personalized information. The experiment's results are used to determine the degree of intimacy between participants. The adjacency matrix of user proximity is formulated at the end of the trial. This paper (Xu X, 2018) presents a methodology that may take into account feedback from both the item and the user community. It employs Ml tools to increase the quality of suggestion in order to strengthen the model's deep learning. Mapped users and things create a representation of the person and the item. Items may be retrieved and ranked using this visual depiction. As a result, the issue is seen as a way to sort things out. To hone the framework, back propagation is employed. Two collaborative models are described by Wu et al. (2019) for the usage of a recommender system. User and item collaborative model strategies are used in this work to design a system that takes use of commonalities across entities Explicit rating refers to how customers rate an item on a certain scale. We can calculate the total number of NN for each user. PCS [2] is used to discover the correlation between user ratings. Rather of focusing on what the item's users enjoy, items focus on what the thing likes. Recommendation is made based on the item's similarity to the target [6].

III Hybrid Approach

Collaborative Filtering (CF), content-based filtering, and knowledge-based filtering all have their advantages and disadvantages. If CF has sparsity and cold start issues, then content-based techniques have narrowness and need descriptions of what they look like. It's possible to create a more reliable recommender system by combining two different approaches.

3.1 Types of Hybrid

Weighted Hybrid: The weighted total of the recommendation ratings for each source is used to calculate a score for each suggested item. The user may adjust the weights for each context source by dragging and dropping on sliders. It is desired, but not straightforward, to automatically optimise the weights for each context source. In order to derive an ideal weighting system, empirical bootstrapping can be utilised, but historical data is required.

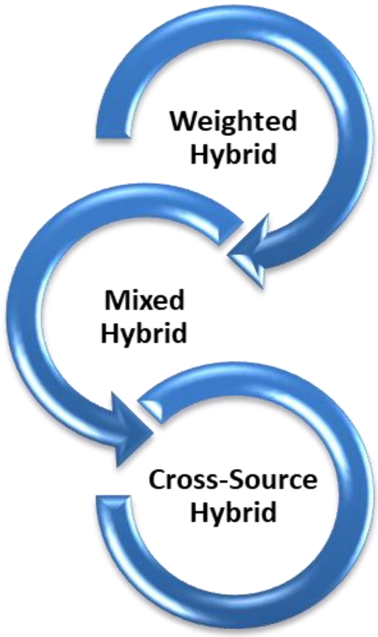


Fig 2: Types of Hybrid

Mixed Hybrid: These recommendations are then sorted by each source and the top-n are selected one at a time by rotating the sources. Individual recommendation ratings are omitted from this technique, which solely evaluates a person's position in a ranked list. Algorithm simply picks the next recommendation from the ranked list if a recommendation is generated from several context sources (i.e. was previously taken from another source).

Cross-Source Hybrid: This method places a high value on recommendations that come from several sources. A suggestion provided by more than one context source / algorithm, such as Facebook's collaborative Filtering and Wikipedia's content-based recommendation, should be regarded as more important [17], according to this study.

3.1.1 Issue with Hybrid Approach

Reliable Integration: The first issue is to make suggestions based on collaborative and content-based data. Collaboration and content-based techniques, either together or separately, may be used in a straightforward manner. This technique, on the other hand, has certain drawbacks. It has been proposed to select a recommended system among traditional ones on the basis of specified quality indicators, however the inadequacies of the selected system are handed down from generation to generation. There is no fundamental rationale for the heuristics-based integration in previous studies [15].

Efficient Calculation: As the number of ratings and users grows, it becomes increasingly difficult for recommender systems to keep up. Memory-based approaches provide a quick and simple solution to this issue because the entire dataset is always used to generate suggestions. Late answers, on the other hand, used a probabilistic technique in an entirely collaborative filtering setting [16] to try to address this shortcoming. On the other hand, an approach for model-based collaborative filtering that gradually trains an aspect model was developed. We are not aware of any studies on incremental adaptation of hybrid recommender systems, thus we cannot comment on them." It's important to think about whether or not past approaches can be used while designing a hybrid architecture.

IV Conclusion

Most of the collaborative filtering, content-based filtering, and hybrid recommendation strategies that have been expected so far have been successful in resolving issues while also giving better suggestions. Nevertheless, with the explosion of information, this study topic must be worked on to discover and create new ways for providing recommendations across a wide variety of applications while taking quality and privacy into consideration. It's now much simpler to track out a good movie thanks to server-based recommendation engines. Assists us identify films that we need to watch instead of searching extensively online and helps cinephiles and movie enthusiasts by recommending top-tier films to watch without digging into vast databases, which is time intensive. For this problem, we propose a collaborative and content-based strategy that uses a range of Machine Learning algorithms from a large database to provide a movie recommendation based on the user's taste and previous viewing history or genre.

References

1. Ahuja R, Solanki A, Nayyar A (2019) Movie recommender system using k-means clustering and k-nearest neighbor. Proc 9th IntConf Cloud Comput Data SciEngConflu 2019 263–268.
2. Indira K, Kavithadevi MK (2019) Efficient machine learning model for movie recommender.pdf
3. Yang D, Zhou Z (2013) Personalized mining of preferred paths based on web log. Proc 2013 IEEE 11th IntConf Electron Meas Instruments, ICEMI 2013 2:993–997.
4. Wu CSM, Garg D, Bhandary U (2019) Movie recommendation system using collaborative filtering. Proc IEEE IntConfSoftwEngServSci ICSESS 2018-November 11–15. https://doi.org/10.1109/ICSESS.2018.8663822
5. Xu X, Zhang Y (2018) Collaborative filtering recommendation algorithm based on hybrid similarity. 2017 IntConfComputSyst Electron Control ICCSEC 2017 2018:1372–1375. https://doi.org/10.1109/ICCSEC.2017.8447000
6. Yang D, Zhou Z (2013) Personalized mining of preferred paths based on web log. Proc 2013 IEEE 11th IntConf Electron Meas Instruments, ICEMI 2013 2:993–997.
7. M. Shah, D. Parikh and B. Deshpande, "Movie Recommendation System Employing Latent Graph Features in Extremely Randomized Trees," in Proceedings of the Second International Conference on Information and Communication Technology for Competitive Strategies (ICTCS '16), New York, 2016.
8. Y. Deldjoo, M. Elahi, M. Quadrana and P. Cremonesi, "Using visual features based on MPEG-7 and deep learning for movie recommendation," Int J Multimed Info Retr, vol. 7, p. 207–219, 2018.
9. J. K. Leung, I. Griva and W. G. Kennedy, "Making Use of Affective Features from Media Content Metadata for Better Movie Recommendation Making," arXiv, 2020.
10. S. Reddy, S. Nalluri, S. Kunisetti, S. Ashok and B. Venkatesh, "Content-Based Movie Recommendation System Using Genre Correlation," in Smart Intelligent Computing and Applications, Singapore, 2019.
11. H. Li, J. Cui, B. Shen and J. Ma, "An intelligent movie recommendation system through group-level sentiment analysis in microblogs," Neurocomputing, vol. 210, pp. 164-173, 2016.
12. P. P. Adikara et al. ISSN 2502-3357 (online) | ISSN 2503-0477 (print) regist. j. ilm. teknol. sist. inf. 7 (1) January 2021 31-42 Movie recommender systems using hybrid model based on graphs with co-rated … http://doi.org/10.26594/register.v7i1.2081 \
13. O.-J. Lee and J. J. Jung, "Explainable Movie Recommendation Systems by using Story-based Similarity," in Explainable Smart Systems 2018 (ExSS '18), Tokyo, 2018.
14. J. Li, W. Xu, W. Wan and J. Sun, "Movie recommendation based on bridging movie feature and user interest," Journal of Computational Science, vol. 26, pp. 128-134, 2018.

15. W. E. Winkler, "String Comparator Metrics and Enhanced Decision Rules in the Fellegi-Sunter Model of Record Linkage," The Educational Resources Information Center (ERIC), Washington, DC, 1990.
16. C. D. Manning, P. Raghavan and H. Schütze, An Introduction to Information Retrieval, Cambridge: Cambridge University Press, 2008.
17. A. Vukotic, N. Watt, T. Abedrabbo, D. Fox and J. Partner, Neo4j in Action, Greenwich, CT, United States: Manning Publications Co, 2014.

An Analytical study on Recommendation systems Using Collaborative Filtering: AHP Perspective

Ismat Anjum [1], Anwar Ahmad Shaikh [2]

[1]CSE Integral University, Lucknow, India

[2]Department of CSE Integral University, Lucknow

Abstract

The recommendation system plays a significant position in the present day and adopted by many popular applications. The recommendation system has established the collection of applications, generating a worldwide community, and increasing for bountiful information. The recommendation system progressed into \sCollaborative Filtering, Content-based, and hybrid-based approaches. It is crucial to supply the user with movie suggestions so that the user does not have to spend a large amount of time looking for material that they would appreciate. As a consequence, the role of the movie recommendation system is highly crucial in order to obtain user-specific movie selections. After doing considerable research on the internet and consulting a large number of scholarly articles, we came to the conclusion that the suggestions generated by Collaborative Filtering only use a single method for converting text to vectors and only use a single method for determining the degree to which vectors are similar to one another. Our project's purpose is to construct a recommendation engine that reacts to the user in order to acquire ideas for a movie.

Index Terms Recommendation Services, Collaborative Filtering, AHP, Comparative analysis

I Introduction

A recommendation system is a form of knowledge filtering system that makes an effort to anticipate the preferences of a user and offers a recommendation based on the user's selections [1]. Systems may perform a range of functions. These have gained popularity in recent years and are now used by most online businesses. Such stages may include movies, music, books, and records, friends and tales shared on online networking media platforms, things sold on online business websites, persons featured on professional and matrimonial websites, and Google search results. Predictive systems are known as recommendation systems because they propose products to users, users to goods, and occasionally even users to users [2]. Recommendation systems also promote users to other users. Tech giants such as YouTube, Amazon Prime, and Netflix all employ similar strategies to propose video material to users based on the interests those users have shown. Because the internet includes vast quantities of data, locating your material may be very challenging and time consuming; hence, recommendations play a crucial part in reducing the amount of effort we have to spend [3]. The majority of the time, these systems will gather data on the decisions made by users and then utilise that data to enhance their recommendations in the future. Adapter, they are able to suggest the Adapter to a first-time customer who has just added a MacBook to their shopping basket. Users are consistently provided with useful suggestions as a result of the advancements made in recommender systems. If a user's preferred genres of music are not available via a music streaming app, the user will likely cease using the service [4, 5]. Because of this, the pressure on technology businesses to improve their recommendation algorithms has increased significantly. However, the problem is bigger than it seems. Every person has their own unique set of preferences and interests. They have to investigate uncharted territories to learn more about the user while also making the most of the information that is currently available on the user. There are three primary strategies that are implemented inside our recommender systems. One of these techniques is called demographic filtering, and it means that the company provides summed-up ideas to each customer according to the predominance of movies or their probable categorization [7]. Customers are given recommendations for films that are comparable to one another in terms of their section highlights. Since every client is diverse, this methodology is viewed as excessively straight forward [8]. This structure is based on the premise that popular films would be enjoyed by the typical audience member. Second, content-based separation, where we profile the client's interests using data and recommend items based on that profile.

II Background

Sang-Min Choi, et. al. [1] discussed the drawbacks of collaborative filtering, such as the cold-start issue or sparsity problem. The authors have come up with a solution to this problem by using category information. A movie recommendation system based on genre correlations has been suggested by the authors. According to the writers, the freshly developed material has category information included in it. Even if a new piece of content doesn't have a lot of ratings or views, it might nevertheless show up in the suggestions list because of its category or genre information. The suggested system is neutral in its treatment of both highly rated and less-watched new material. Because of this, even a newly released film might be suggested by the recommendation system. **George Lekakos, et. al.** [2] presented a hybrid approach to movie recommendation as a solution. According to the authors, both Content-based filtering and Collaborative filtering have advantages and disadvantages. A hybrid strategy has been devised by the authors, which takes into account both content-based and collaborative filtering techniques. A movie recommendation system known as 'MoRe' uses the solution. The Pearson correlation coefficient was not employed for the purpose of collaborative filtering. In place of this, a new formula has been used. However, there is an error in this calculation that results in a "division by zero." This error takes place if there is a deadlock in the ratings for a movie. As a direct consequence of this, the authors failed to take into account these users. Within the context of a recommendation system that is only focused on the content of the movie, the authors have made use of cosine similarity to take into consideration the film's writers, actors, directors, and producers, in addition to the genre. They developed a hybrid recommendation approach by combining two different strategies, which they referred to as "substituting" and "switching." Both of these systems make use of collaborative filtering as well as content-based filtering, both of which are designed to provide results only when certain conditions are met. Because of this, the authors use the collaborative filtering methodology as their primary method. **Debashis Das, et. al.** [3] This article discusses the many kinds of recommendation systems and provides basic information about them. An overview of recommendation systems was provided in this work. The authors discussed customised and non-personalized recommendation systems. An excellent example was used to demonstrate the difference between user-based and item-based collaborative filtering. Various recommendation systems have also been discussed by the writers.

III Collaborative Filtering (CF)

It merely advertises the items to those with similar tastes by filtering out material based on user interest that is comparable to that of other users. It is also a well-known algorithm in a variety of fields. There are two primary filtering algorithms in memory-based techniques. Model-based techniques, on the other hand, are less reliable than memory-based tactics. In the event that sufficient data is available, collaborative filtering-based recommendation systems may provide an accurate prognosis since they are based on the user's preferences. When it comes to predicting consumer behaviour, the most critical part of a recommendation system, user-based collaborative filtering has proven extremely successful in the past. In spite of this, their widespread use has exposed certain real challenges, such as data sparsity and data scalability, as the number of users and items continues to grow. It is necessary to have a collection of items that are reliant on the user's previous choices in order to employ collaborative filtering. This approach does not need a substantial quantity of product features to function. An embedding or feature vector represents each item and User, and it sinks both the things and the users in a same embedding position. It develops enclosures for goods and users on its own. Other purchaser’s responses are taken into account when offering a certain product to the principal user. It keeps track of the behaviour of all users before proposing which item is generally loved by people. It also links comparable consumers by similarity in desire and behaviour towards a similar product when suggesting a product to the core client. This chapter suggested item-based collaborative filtering as an application of dimension reduction in a recommendation system so that the prediction problem's execution time could be shortened and its accuracy could be increased. It illustrates that the suggested method is capable of achieving higher performance and execution time for the recommendation system in terms of the current problems, as shown by validation via AHP approaches. In figure 1, we have shown the flow structure of the research work behaviour from a number of different points of view.

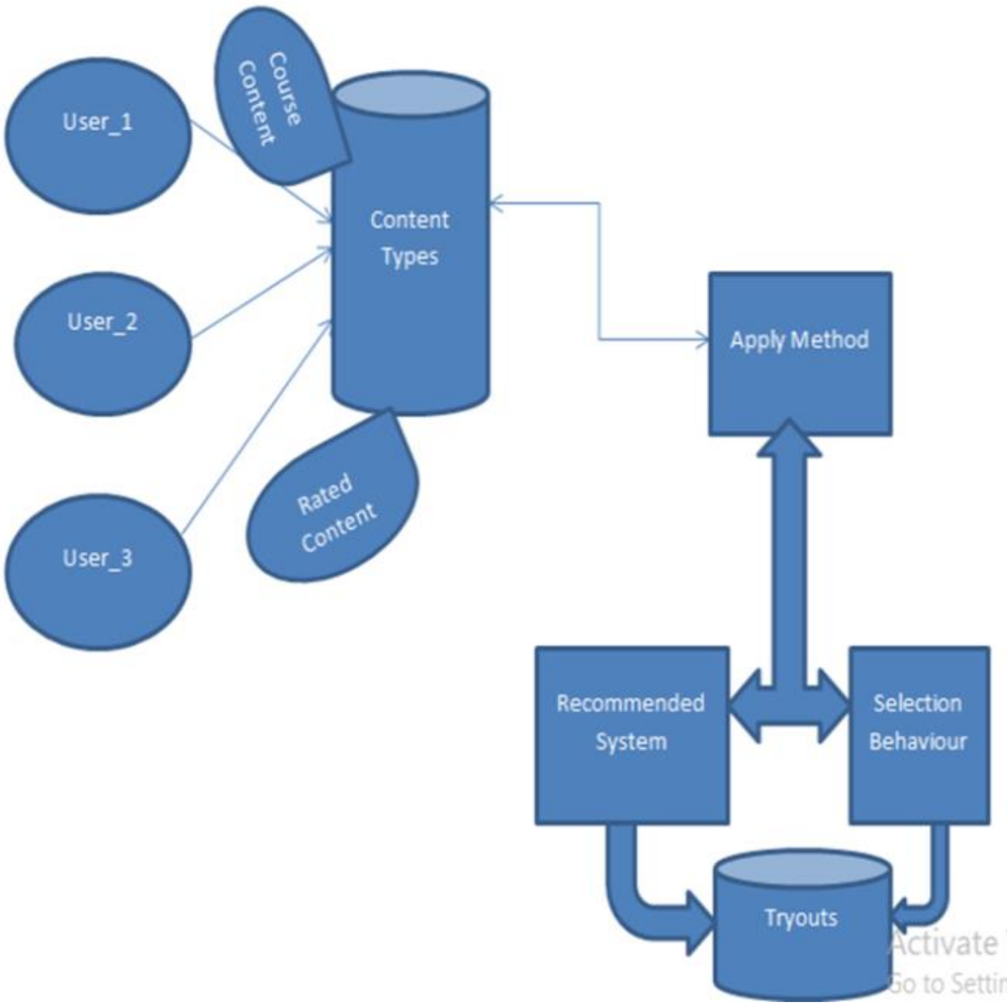


Figure 1: Flow Structure of Research

3.1 Computing Similarity among Users

In order to determine the degree of similarity between users, we use the idea of Pearson correlation, as well as the cosine similarity and user-based prediction computing formula.

3.1.1 Pearson Correlation

In order to determine the degree of similarity between two users' ratings, this technique calculates the statistical correlation between those users' evaluations using “Pearson's r”. The following formula is used to determine the correlation:

sim (i,j)= $\frac{\Sigma u \in U(Ru,i-\bar{R}u)(Ru,j-\bar{R}u)}{\sqrt{\Sigma u \in U(Ru,i-\bar{R}u)^2}\sqrt{\Sigma u \in U(Ru,j-\bar{R}u)^2}}$.

(Eq. 2.1)

Where,

i: Set of items rated by the user.

R_u, i: Is the rating given to item I by user (u).

$\bar{R}_u$: Is the mean rating given by user (u).

Cosine Similarity

Nearest-Neighbor CF algorithm:

A measure of the degree of similarity between two non-zero vectors of an inner product space is referred to as the cosine similarity, and it is determined by the cosine of the angle that separates them. The cosine of an angle of 0 degrees equals 1, but the cosine of every other angle is less than 1.

For N-dimensional vector of items, measure two customers A and B

Similarity ($\vec{A}$,$\vec{B}$)= cos($\vec{A}$,$\vec{B}$)= $\frac{\vec{A}.\vec{B}}{||(\vec{A})||*||(\vec{B})||}$ (Eq. 2.2)

When calculating cosine similarity, negative ratings are not allowed, and unrated things are given a rating of zero regardless of whether or not they have any ratings at all.

2.3.3 User Based Prediction Computing Formula

$p_{C,e}$= $\bar{r}C + \frac{s(C,A)(rA,e-\bar{r}A)+s(C,D)(rD,e-\bar{r}D)}{(|s(C,A)|+|s(C,D)|)}$ (Eq. 2.3)

(pC,e): User C’s prediction for Equilibrium

$\bar{r}C$: Mean rating of user C

$\bar{r}A$: Mean rating of user A

$\bar{r}D$: Mean rating of user D

s(C,A): similar user A and C

s(C,D): similar user C and D

From above formula we find the prediction of user in unknown item. Due to this, then we recommend this item to that user whose was not rated this item. Now we consider a table of ratings of different users.

Table 1 Ratings matrix

	Item 1	Item 2	Item 3	Item 4
User A	4	?	3	5
User B	?	5	4	?
User C	5	4	2	?
User D	2	4	?	3
User E	3	4	5	?

Take a look at the ratings matrix in Table 1, and keep in mind that we need to locate User C's estimate for the value of Equilibrium (pC,e) using the following configuration:

• Pearson correlation.

• Neighborhood size of 2.

$\bar{r}C = \frac{(5+4+2)}{3}$

$\bar{r}C = 3.667$

C’s average score is 3.667. There are only two users who have given the game a rating, and as a result, there are only two people who are candidates for inclusion in the neighbourhood:

$$s(C,A)=\frac{(4-4)*(5-3.67)+(3-4)(2-3.67)}{\sqrt{((0)^2+(-1)^2)}*\sqrt{((1.33)^2+(-1.67)^2)}}$$

$$=0.784$$

A and D, *s*(*C,A*) = 0.784 and *s*(*C,D*) = −0.518 from Equation 1.1.The prediction pC,e is therefore computed as follows:

$$p_{C,e}=\bar{r}C+\frac{s(C,A)(rA,e-\bar{r}A)+s(C,D)(rD,e-\bar{r}D)}{(|s(C,A)|+|s(C,D)|)}$$

$$= 3.67+\frac{0.784*(5-4)+-0.518*(2-3)}{0.784+0.518} = 4.667$$

Here we calculated the value of user rating of C. On the behalf of this rating we can recommend items to the users.

IV Impact Analysis

An accurate assessment of the relevant data is one of the procedures that is considered to be among the most critical in several decision-making processes. This is a difficulty that is not exclusive to the AHP technique alone; rather, it is essential in a wide variety of other approaches that need the decision-maker to provide qualitative information. There is a high likelihood that qualitative data cannot be known in terms of absolute values and are not unique. As a result, a significant portion of AHP is dedicated to figuring out the relative relevance, or weight, of the various options in terms of each criteria that is associated with a specific decision-making issue. With the Analytic Hierarchy Process (AHP), pairwise comparisons are conducted to establish the relative relevance of each choice in relation to each criteria of the django framework. In this method, the individual in charge of making decisions is tasked with providing commentary on the significance of a single comparative pair at a time. In most cases, the person making the decision will be required to choose his response from a number of distinct options. In order to put this validation into action, we must begin by taking the parameters, whose numeric values represent the heading and velocity. The prediction model provides an explicit estimate of the heading, while the point model delivers an implicit estimate of the heading depending on the selection. After that, we take a model of the movie system that was suggested to us. In order to compute, we make advantage of the many domains that the dynamic paramerts provide. The next thing we do is form the hypothesis that the dynamic model will go on with its scenario, and that it will most likely keep the same offset that it has now. The researchers will get information on the fundamental steps involved in the recommendation system.

The recommendation system operates mostly based on the specifics of the product as well as the data of the user. We are required to get them from the system or from the database, and based on the ratings, we will make our selections. If an item that is comparable to what was searched for was discovered, then a recommendation system will be developed; otherwise, no recommendation system will be formed.

4.1 AHP Methodology:

When it comes to making decisions, the Analytic Hierarchy Process (AHP) was developed by Saaty (1977 and 1994). Many scholars are intrigued by the AHP, in part because of its appealing mathematical features and the relative ease with which it may be used with readily available input data. Using the AHP as a decision support tool, you may work through more difficult issues of decision-making. Objectives, criteria, sub-criteria, and options are organised into a multi-level hierarchical framework. A series of pairwise comparisons yields the relevant information. For each individual choice criterion, these comparisons are utilised to determine how important that criterion is, as well as how well each option performs in contrast to the others. In cases when the comparisons aren't exactly the same, this is a way to make things more consistent going forward. In table 5.1, Let C = {Cj| j = 1, 2... n} be the set of decision criteria. The data of the pair wise comparison of n sub-criteria can be summarized in an (n× n) evaluation matrix A in which every element aij (i, j = 1, 2 ... n) is the quotient of weights of the criteria. A square matrix and a reciprocal matrix may be used to illustrate this pair-wise comparison. In this matrix $a_{ij} = 1/a_{ji}$, for all experts, we would have (n× n) matrices (see table 5.1).

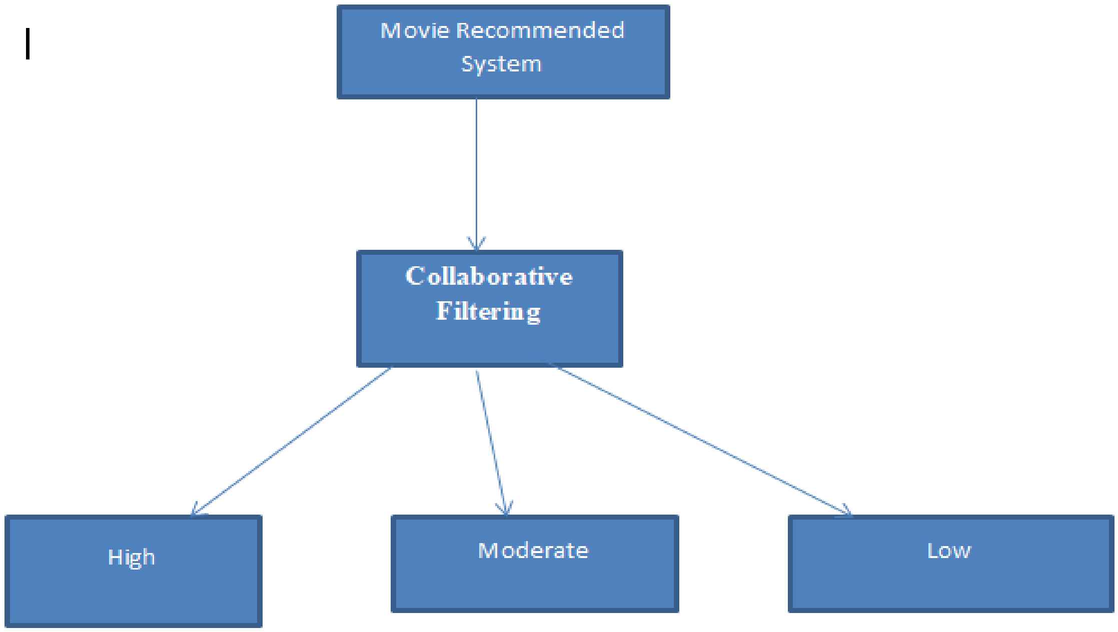


Figure 2: Conceptual Behaviour

Table 1 Assign Weight (Collaborative Filtering)			
	High	**Low**	**Moderate**
High	1	3.9	5.6
Low	0.2564	1	3.2
Moderate	0.1785	0.3125	1
	1.435	5.213	9.800

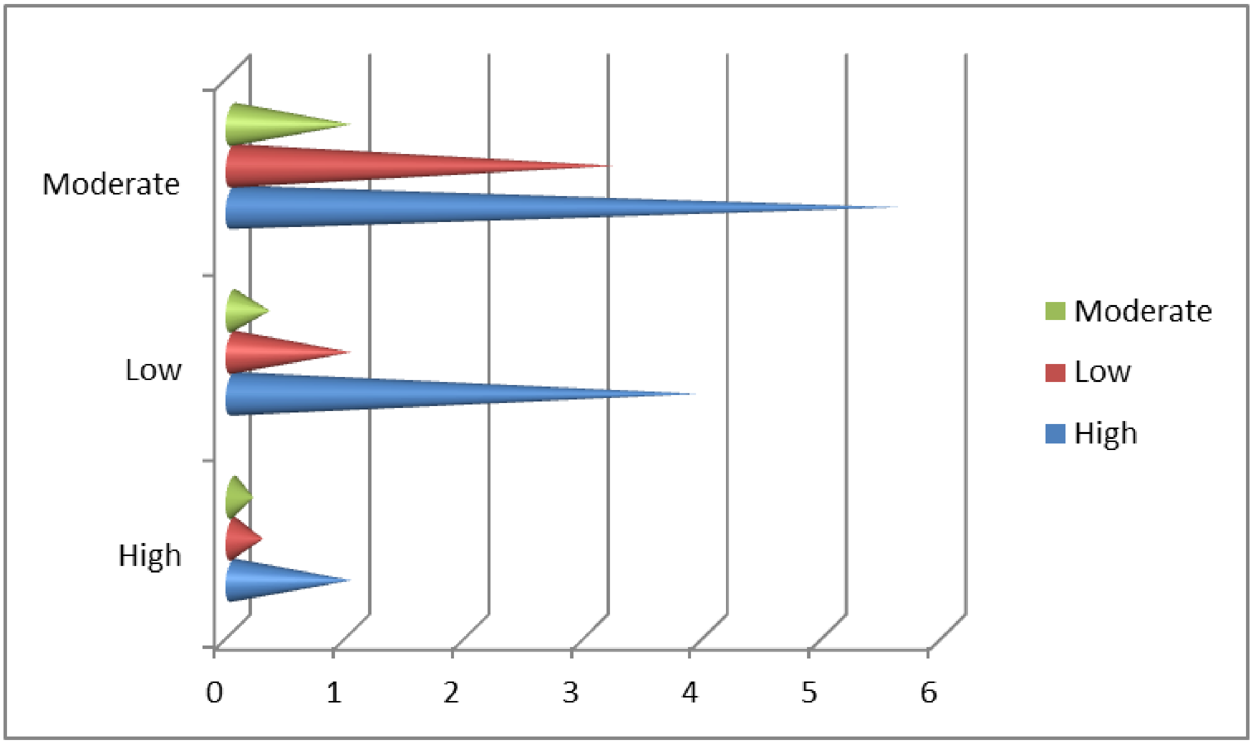


Figure 3: Graphical Valuation

In Table 2, the Accuracy resulted in the best value with an ideal priority Vector 0.6722, mainly because it was the highest evaluated in the two metrics: 0.6969 and 0.7482 (figure 3). Nonetheless, there are three alternatives for model that also stand out. In table 2, it is possible to see the average consistency index obtained from the output of the alternatives in the test phase.

As a result of the curse of dimensionality, it is possible to use the AHP to calculate the options among different models and justify the model's accuracy. The new approach has been introduced to solve the most important alternatives, and the details of this approach are provided in Table 2 with consistency index for verifying the stage calculations.

Table 2 Normalized Metrics

	High	**Low**	**Moderate**
High	0.6969	0.7482	0.5714
Moderate	0.1787	0.1918	0.3265

Low	0.1244	0.0600	0.1020
Eigen Vector		**Priority**	
0.9645		0.6722	
1.2111		0.2324	
0.9357		0.0955	
Eigen Value 3.114		0.0557	

In table 3, we have calculated to overall priority of each criteria respect to model weight. We are observed that high values of very effective in this research work. Table 3 are given the finalize metrics in the form of priority High Low Moderate context. Accuracy value is the maximum effective constraints which provides verifying and validated of research.

Table 3 Calculate overall priority

	High	**Low**	**Moderate**
High	0.6722	0.653	0.600
Moderate	0.2324	0.251	0.200
Low	0.0955	0.096	0.200

Table 4 Finalize Metrics

High	**0.6252**
Moderate	0.2137
Low	0.1611
Highest Priority = Highest Score	

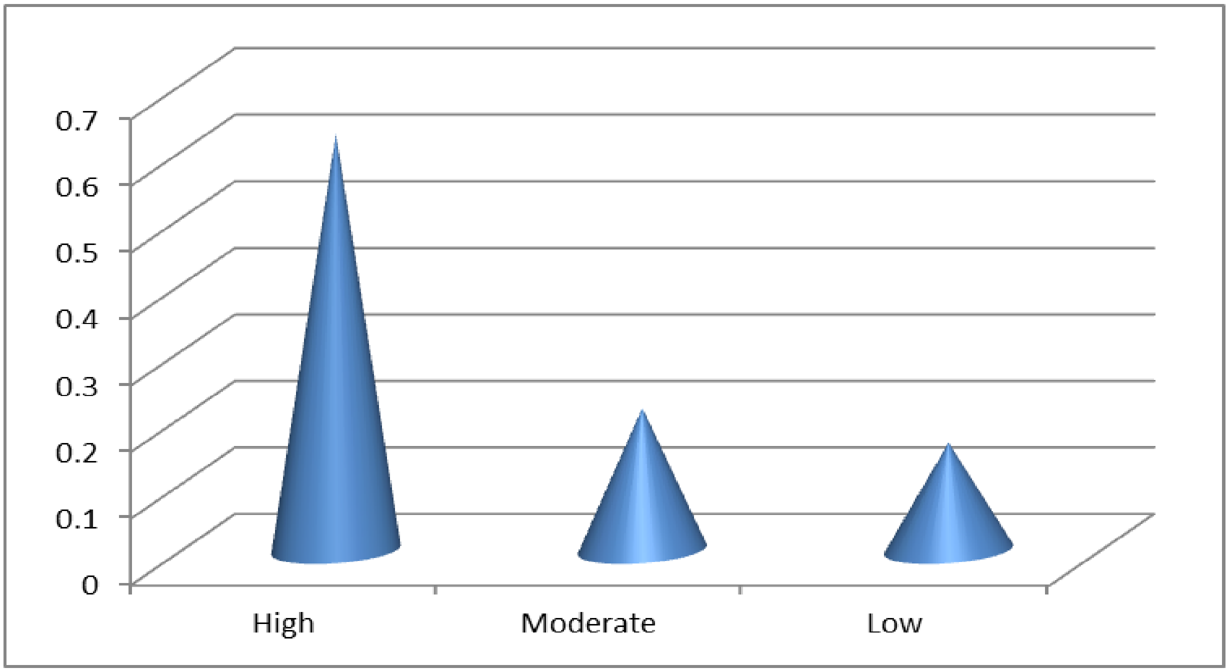


Figure 4: Final Structure

The inconsistency of the AHP pairwise comparison matrix is addressed in this study using three different models, depending on three different levels of inconsistency: high, low, and moderate. Initially, both methodologies are compared using simulations that include training, validation, and testing. Compared to High Value in CR reduction, the Collaborative Filtering approach has a similar behaviour but is better at anticipating previously unknown inputs supplied to the network, giving it a major advantage over the former. It's possible to observe the ultimate AHP weight in Figure 4 (right).

5. Comparative analysis

After we have obtained the findings, it is strongly advised that you compare the work that was proposed with other recent work that has been done in the same area. Manoj et al. (2015) have implemented a method for a movie recommendation system that is based on the weighting of criteria, and this method is quite similar to the one that we have suggested here. In the future, we want to concentrate on improving its user interface as well as its weaknesses.

Table 1 Comparison of ranks between proposed work and existing work (Manoj et.al(2015))

	Weight of Proposed work	Ranking of Proposed work	Weight of Existing work	Ranking of Existing work
Aggregate	**0.6252**	1	0.60	1
	0.2137	2	0.54	2
	0.1611	3	0.48	3
Moderate	0.653	1	0.42	3
	0.251	2	0.36	2
	0.096	3	0.30	1
High	0.6722	1	0.61	3
	0.2324	2	0.48	2
	0.0955	3	0.42	1
Low	0.600	1	0.45	3
	0.200	2	0.36	2
	0.200	3	0.06	1

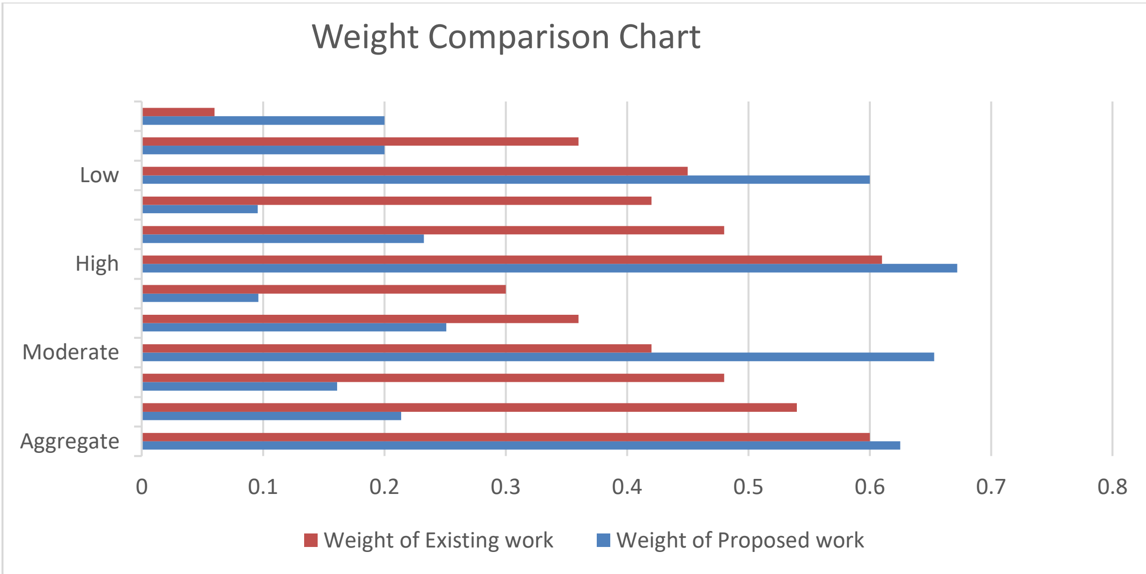


Figure 5: Weight Comparison Chart

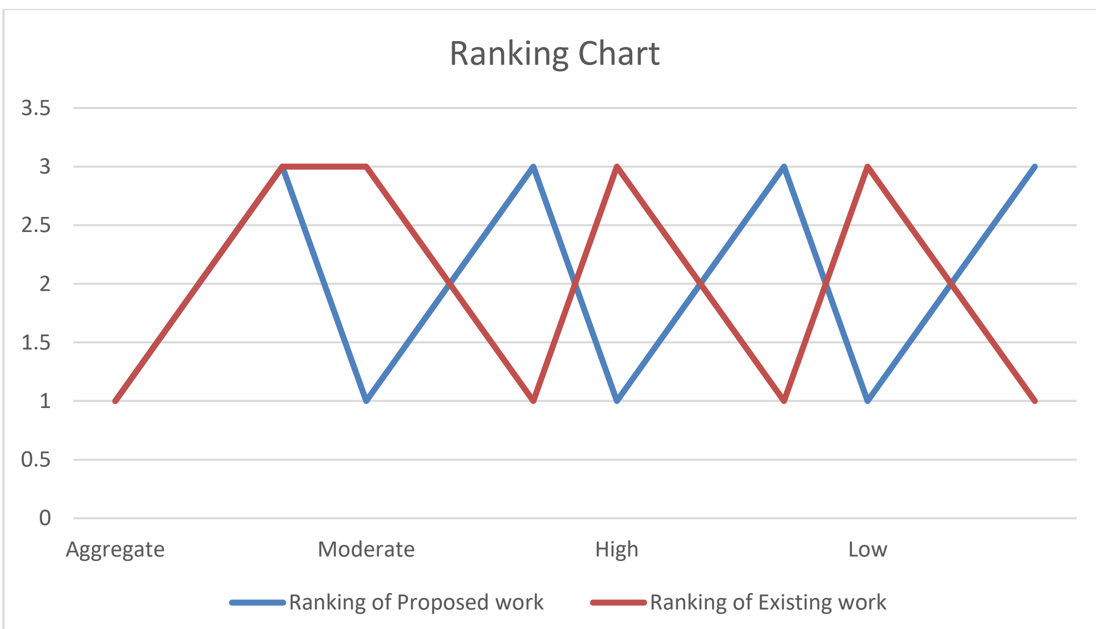


Figure 6: Rank Comparison Chart

6. Conclusions

There are new routes for obtaining personal information online thanks to the advised solutions. A common problem with information retrieval systems is an overabundance of information, and this solution helps to relieve that problem while simultaneously providing customers with quick, convenient access to products and services that may otherwise be unavailable. Based on the input that customers have given us in the past and our strategy to respond to their unique needs and interests, we recommend movies to them. As a result of this method, known as AHP, the recommendations model becomes more accurate. In addition, the system is more responsive and the data collecting procedure is more accurate as a result of this. In addition to confirming the model, this approach also improve the system's performance. When it comes to discovering new things, the Recommendation system may be a lifesaver. Certain restrictions prevent these systems from suggesting effectively to their consumers. Even though Collaborative Filtering is the most effective and powerful method, this algorithm has a high runtime and confronts challenges like as data sparsity, which may be eliminated by employing a Hybrid movie recommendation system. However, there are advantages and disadvantages to each option. In our suggested validation method, we can deal with quite a large quantity of data with ease. We plan to address its weaknesses and improve its user interface in the future.

References

1. Choi, Sang-Min, Sang-Ki Ko, and Yo-Sub Han. "A movie recommendation algorithm based on genre correlations." Expert Systems with Applications 39.9 (2012): 8079-8085.
2. Lekakos, George, and Petros Caravelas. "A hybrid approach for movie recommendation." Multimedia tools and applications 36.1 (2008): 55-70.
3. Das, Debashis, Laxman Sahoo, and Sujoy Datta. "A survey on recommendation system." International Journal of Computer Applications 160.7 (2017).
4. Dietmar Jannach, Gerhard Friedrich; “Tutorial: Recommender Systems”, International Joint Conference on Artificial Intelligence, Beijing, August 4, 2013.
5. Gaurangi, Eyrun, Nan; “Movie GEN: A Movie Recommendation System”, UCSB.

6. Harpreet Kaur Virk, Er. Maninder Singh,” Analysis and Design of Hybrid Online Movie Recommender System “International Journal of Innovations in Engineering and Technology (IJIET)Volume 5 Issue 2,April 2015.
7. Manoj Kumar, D.KYadav, Ankur Singh, Vijay Kr. Gupta,” A Movie Recommender System: MOVREC” International Journal of Computer Applications (0975 –8887) Volume 124 – No.3, August 2015.
8. Prerana Khurana , Shabnam Parveen; ‘Approaches of Recommender System: A Survey’; International Journal of Computer Trends and Technology (IJCTT) Volume 34 Number 3 - April 2016.