



Movie Recommendation System Based on a Hybrid Approach

Yasir Riaz, Ujjwal Deep and Ankit Darad

EasyChair preprints are intended for rapid dissemination of research results and are integrated with the rest of EasyChair.

July 11, 2021

Movie Recommendation System Based on a Hybrid Approach

Yasir Riaz

School Of Computer Science and

Engineering

Galgotias University

Greater Noida, India

yasirriaz001@gmail.com

Ujjwal Deep

School Of Computer Science and

Engineering

Galgotias University

Greater Noida, India

ujjwaldeep1111@gmail.com

Ankit Darad

School Of Computer Science and

Engineering

Galgotias University

Greater Noida, India

ankitdarad.ad@gmail.com

Keywords: Recommendation Systems, Collaborative filtering, Model-based, Memory-based machine learning, Model-based methods, user-based collaborative filtering, Item-based collaborative filtering, JAVAEE system, KNN algorithm

Abstract—

A recommendation engine filters the data using different algorithms and recommends the most relevant item to users. It first captures the past behaviour of a customer and based on that, recommends products which the users might be likely to buy or watch. If a completely new user visits an e-commerce site, that site will not have any past history of that user. The possible solutions for this could be to tell the best selling products, i.e. the products which are high in demand or to recommend the products which would bring the maximum profit to the business. Three main approaches are used for recommender systems. One is Demographic Filtering i.e. They offer generalised recommendations to every user, based on movie popularity and/or genre. Second is content based filtering, where we try to profile the user's interests using information collected, and recommend items based on that profile. The other is collaborative filtering, where we try to group similar users together and use information about the group to make recommendations to the user. Personalised recommendation system can play an important role especially when the user has no clear target movie. In this paper, we design and implement a movie recommendation system prototype combined with the actual needs of movie recommendation through research in KNN algorithm and collaborative filtering algorithm.

1. INTRODUCTION

Our Movie Recommendation System aims at developing a Software best suited for Platform Streaming Services such as Netflix, Prime Video, Sony LIV, Facebook, etc. We plan to apply Machine Learning using Python Programming Language to develop something best suited for these platforms, which will not only help the users with better recommendations, but it will also allow them to have a broader range of movies presented and recommended to them for watching. This will automatically enhance the quality of the platform, making it attractive for the end users and better to use. Often, these systems are able to collect information about the users' choices, and can

use this information to improve their suggestions in the future. For example, Facebook can monitor your interaction with various stories on your feed in order to learn what types of stories appeal to you. Sometimes, the recommender systems can make improvements based on the activities of a large number of people. For example, if Amazon observes that a large number of customers who buy the latest Apple Mac book also buy a USB Adapter, they can recommend the Adapter to a new user who has just added a Macbook to his cart. Personalised recommendation system is a kind of information filtering technology. It is an integrated system which is a combination of a variety of data mining algorithms and user related information, to meet the interests or potential interests of users. The common recommendation system is categorised as content based recommendation system, collaborative filtering recommendation system, and hybrid recommendation system [9,10]. Each recommendation algorithm has different use range and use condition, it results in the use of different recommendation algorithm for the same information recommendation. In the actual application of recommendation system, the system tends to be a hybrid recommendation system. That is, to mix the advantage of each recommendation algorithm to the recommended process to effectively improve the recommendation effect. In this paper, the key research content is to help users to obtain user interested movie automatically in the massive movie information data using KNN algorithm and collaborative filtering algorithm, and to develop a prototype of movie recommendation system based on KNN collaborative filtering algorithm.

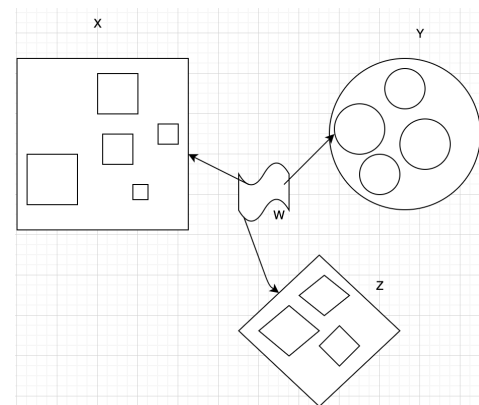


Fig.1- Example of KNN Algorithm

2. LITERATURE SURVEY

2.1.KNN algorithm

KNN algorithm is called K nearest neighbour classification algorithm. The core idea of the KNN algorithm is if the majority of the k most similar neighbours of a sample in the feature space belongs to a certain category, then the sample is considered to belong to this category. As shown in Figure 1, the majority of w's nearest neighbours belong to the x category, w belongs to the X category.

2.2.Collaborative Filtering Based Systems-

Our content based engine suffers from some severe limitations. It is only capable of suggesting movies which are close to a certain movie. That is, it is not capable of capturing tastes and providing recommendations across genres. Also, the engine that we built is not really personal in that it doesn't capture the personal tastes and biases of a user. Anyone querying our engine for recommendations based on a movie will receive the same recommendation for that movie, regardless of who she/he is. Therefore, in this section, we will use a technique called Collaborative Filtering to make recommendations to Movie Watchers.

It is basically of two types:-

User based filtering -

These systems recommend products to a user that similar users have liked. For measuring the similarity between two users we can either use person correlation or cosine similarity. This filtering technique can be illustrated with an example.

Item Based Collaborative Filtering -

Instead of measuring the similarity between users, the itembased CF recommends items based on their similarity with the items that the target user rated. The major difference is that, with itembased collaborative filtering, we fill in the blank vertically, as oppose to the horizontal manner that userbased CF does. It successfully avoids the problem posed by dynamic user preference as itembased CF is more static. However, several problems remain for this method. First, the main issue is scalability. The computation grows with both the customer and the product. The worst case complexity is $O(m \cdot n)$ with m users and n items. In addition, sparsity is another concern.

3. PROBLEM FORMULATION

With complete determination and a concrete research on the subject, the solution to come up with the most competent and modern Movie Recommendation System was successful. Our Movie Recommendation System aims at developing a Software best suited for Platform Streaming Services such as Netflix, PrimeVideo, SonyLIV, Facebook, etc. A hybrid approach can be taken between context based filtering and collaborative filtering to implement the system. This approach overcomes drawbacks of each individual algorithm and improves the performance of the system. Techniques like Clustering, Similarity and Classification are used to get better recommendations thus increasing precision and accuracy. We plan to apply Machine Learning using Python Programming Language to develop something best suited for these platforms, which will not only help the users with better Recommendations, but also allow them to

have a broader range of movies presented and recommended to them for watching. This will automatically enhance the quality of the platform, making it attractive for the end users and better to use.

4. PROPOSED SYSTEM

This Recommendation system will have the solution for all the problems and challenges, once and for all concerning its safety of data and the correct usage of it by providing useful recommendations to the users, hence enhancing the system and the platform. This system is majorly solved and provided the users a better experience on different platforms for watching movies by not making them manually select any movie every time they wish to watch. The system will put the movies into different categories such as horror, comedy, romance, etc and according to the user's preferences, recommends the best movies based on the user's history. In case the user signs up for the first time, he shall be provided the best sellers, or top 10 movies being watched on the platform by an Artificially Intelligent System on a screen which will help him. The System will recommend according to the category of the movies, and thus, efficiently manages the time of the user.

5. KNN COLLABORATIVE FILTERING ALGORITHM

KNN collaborative filtering algorithm, which is a collaborative filtering algorithm combined with KNN algorithm, use KNN algorithm to select neighbours. The basic steps of the algorithm are user similarity calculation, KNN nearest neighbour selection and predict score calculation. In user similarity calculation, The similarity between users is calculated by evaluating the value of the items evaluated by two users. Then comes the KNN nearest neighbour selection, in which the algorithm selects a number of users the highest similarity as the U's neighbour, denoted as u' . After determining the user's neighbours, the score can be predicted according to the score of the neighbour to the item, which is known as predict score calculation.

6. RECOMMENDATION SYSTEM DESIGN

6.1. Main Objective

We aim at developing a Software best suited for Platform Streaming Services such as Netflix, PrimeVideo, SonyLIV, Facebook, etc.

6.2. Specific Objectives

This Recommendation Systems has the following objectives To develop a Software that works as a Recommendation System for Movie best suited for Platform Streaming Services such as Netflix, Prime Video, etc, to apply Machine Learning using Python Programming Language to develop something best suited for these platforms, which will not only help the users with better Recommendations, but also allow them to have a broader range of movies presented and recommended to them for watching and to make a good research about Recommendation Systems and their working, hence gather all necessary information that helps in designing the new movie recommendation system.

6.3. Architecture Design

There are three components in the system, including movies, users and the recommendation system. The recom

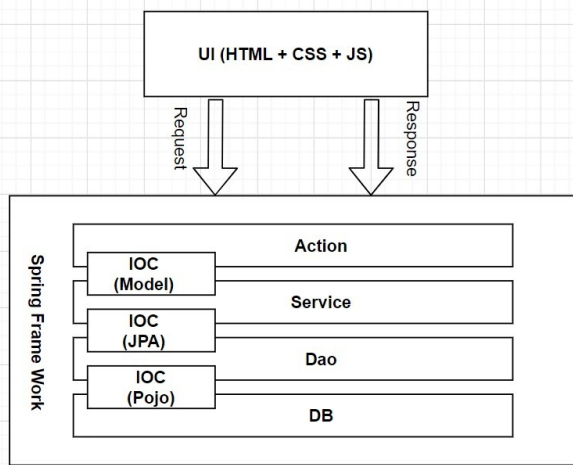


Fig. 2. System Architecture

mentation system determines the dynamic movies based on real-time movie and user related information, and broadcast recommendable prices to users. The price reflects the relationship between demand and supply. Upon receiving different types of recommendations, the user selects a desired movie he wants to watch and puts in the watch later list if he does not wish to watch at that time.

6.4. Database Design

Database is the basis of the system, this system uses MySQL database, the overall database structure diagram is shown in the following figure 3, representing their entity constraints between the data tables [18]. Table Users is the description of user information, including user ID, user name, password, registration time, etc. Table UserSimilar is the description of the user similarity information, including the user similarity ID, user ID, similar neighbour user ID, and the value of the similarity. Table Score is the description of users' rating information on the film, which is the direct information source of collaborative filtering algorithm, it includes the score ID, the user's ID who give the score, the value of the score, content of comments. Table Movie is the description of the movie information, including the movie ID, movie name, director, movie URL, etc. Table MovieType is the description of type information of the movies, including the ID of movies' type, movie name, and type ID. Table MovieSimilar is the description of the movie similarity information, including the movie similarity ID, movie ID, the ID of highly similar neighbour, the value of similarity. Both the table UserSimilar and table Movie similar are the basis of the recommendation algorithm and system [15,16].

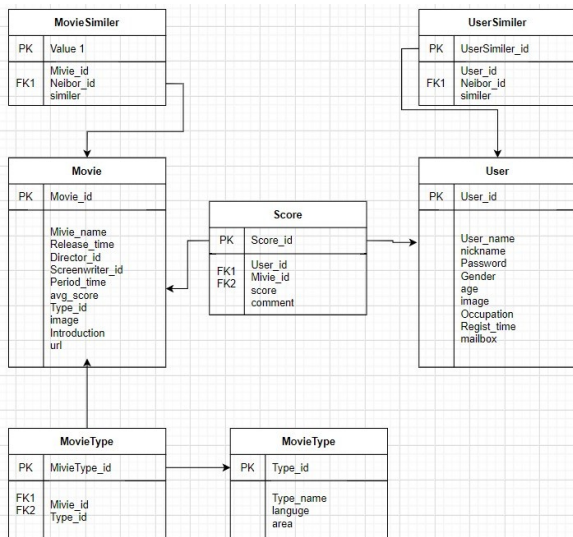
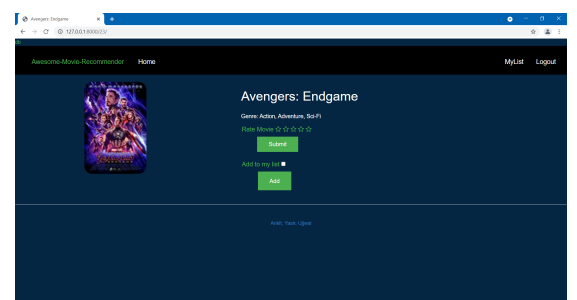
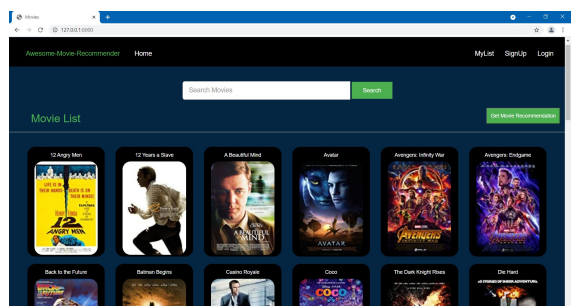
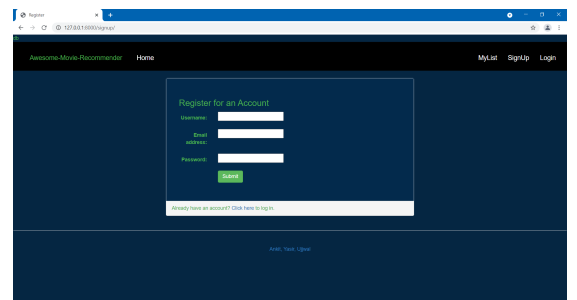
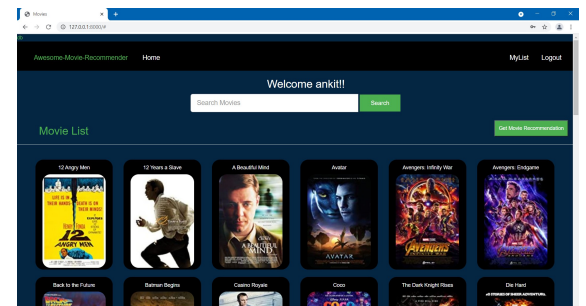
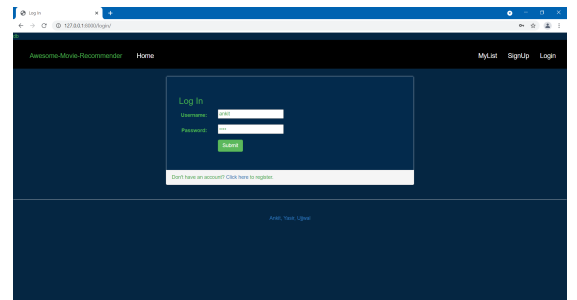


Fig. 3. Relational Model of Database

7. IMPLEMENTATION

User registration system will capture the user's explicit and implicit behavioural characteristics and these characteristics are stored in the user database through the user login module. After logging in to the system, the system will make the appropriate recommendation according to the user's information [19,20]. As shown in all the figures below.



Figs- Display of the implemented recommendation system

CONCLUSION AND FUTURE SCOPE

A hybrid methodology ought to be taken between KNN based separating and community oriented sifting to work the framework. This methodology will dispose of disadvantages of every calculation and will make the presentation of the framework all the more better. Strategies like Clustering, Similarity and Classification are utilised to improve proposals consequently expanding exactness and precision. In future we can work on hybrid recommender using clustering and similarity for better performance.

- Our approach can even be further extended to other domains to recommend songs, video, venue, news, books, tourism and e-commerce sites, etc.
- The features provided by use for more interactive enhancement of the screens and inclusion of more data.

REFERENCES

1. "Movie Recommendation System", a Software Design by Yasir Riaz, Ankit Darad and Ujjwal Deep
2. "Commonly Used Machine Learning Algorithms", <https://www.analyticsvidhya.com/blog/2017/09/common-machine-learning-algorithms/>
3. Netflix, for guidance
4. Sri, M. N., Abhilash, P., Avinash, K., Rakesh, S., & Prakash, C. S. (2018). Movie Recommender System using Item-based Collaborative Filtering Technique.
5. <https://www.javatpoint.com/k-nearest-neighbor-algorithm-for-machine-learning>
6. <https://realpython.com/build-recommendation-engine-collaborative-filtering/>
7. Collaborative Filtering, on Wikipedia
8. "Commonly Used Machine Learning Algorithms", <https://www.analyticsvidhya.com/blog/2017/09/common-machine-learning-algorithms/>
9. "Investigating the use of Machine Learning for Smart Parking Applications" by Jonathan Baker and Sabih Ur Rehman
10. k-nearest neighbours algorithm, on Wikipedia