MOVIE RECOMMENDER SYSTEM USING COLLABORATIVE FILTERING

Submitted by

MUSSA NATHANIEL (2K20CSE26)

Submitted in partial fulfillment of the requirement for the degree of

MASTER OF COMPUTER SCIENCE AND ENGINEERING

Under the guidance of

Dr. MANOJ KUMAR E.

A PROJECT REPORT

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING,

DELHI TECHNOLOGICAL UNIVERSITY.

AKNOWLEDGEMENT

I would like to take this opportunity to express my gratitude regards to my mentor Dr. Manoj Kumar for providing me with this opportunity, for his guidance, monitoring and constant encouragement throughout the development of this project. This project would not have been possible without his innovative sight and practical approach towards learning. Moreover, I would like to express my regards to our institution. The culture and environment of Delhi Technological University is perfect when it comes to projects and hands-on learning. Finally, I would also express my thanks to various colleagues, who helped me at many stages to overcome a lot of hindrances and helped me in understanding various aspects of the project.

ABSTRUCT

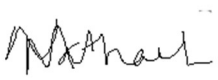
Abstract: Movies are one of the sources of entertainment, but the problem is how to find what you need from the millions of contents that is increasing every year. However, in these cases, the recommendation system is much more convenient. The purpose of this article is to improve the accuracy and performance of conventional filtering techniques. Although multiple methods are used to implement a recommendation system, content-based filtering is the simplest method. It accepts the user's input, rechecks his/her history/past behaviour, and recommends a list of similar movies. In this article, in order to prove the effectiveness, compared with content-based filtering, K-NN algorithm and collaborative filtering mainly focus on improving the accuracy of the results. Cosine similarity is used as the accuracy of cosine angle and the equidistance of movies remain almost the same.

Keywords— Movie recommender system, cosine similarity, KNN algorithms, content-based filtering, collaborative filtering, nearest neighbours

DECLARATION BY THE CANDIDATE

I **Mr. MUSSA NATHANIEL (2k20cse26)** hereby declare that this Project Work report **"ONLINE HOTEL BOOKING SYSTEM"** submitted to **Department of Computer Science and Engineering, Delhi Technological University.** in the partial fulfillment of requirements for the Master Degree in Computer Science and Engineering, This Project record work done by me under the supervision of **Mr. Manoj Kumar.**

Signature of the Student



MUSSA NATHANIEL

CERTIFICATE OF THE GUIDE

This is to certify that the Project work entitled **"ONLINE HOTEL BOOKING SYSTEM"** is a Bonafede work of **Mr. MUSSA NATHANIEL,** Registration No**. ()2K20CSE26** in partial fulfillment for the award of the degree of **Master in Computer Science and Engineering** of **Delhi Technological University** under my guidance. This work is original one and not submitted earlier for the award of any degree elsewhere.

Dr. MANOJ KUMAR
PROJECT GUIDE
DEPARTMENT OF COMPUTER SCIENCE
DELHI TECHNOLOGICAL UNIVERSITY

LIST OF CONTENTS

CHAPTER ONE

Abstract: Movies are one of the sources of entertainment, but the problem is how to find what you need from the millions of contents that is increasing every year. However, in these cases, the recommendation system is much more convenient. The purpose of this article is to improve the accuracy and performance of conventional filtering techniques. Although multiple methods are used to implement a recommendation system, content-based filtering is the simplest method. It accepts the user's input, rechecks his/her history/past behaviour, and recommends a list of similar movies. In this article, in order to prove the effectiveness, compared with content-based filtering, K-NN algorithm and collaborative filtering mainly focus on improving the accuracy of the results. Cosine similarity is used as the accuracy of cosine angle and the equidistance of movies remain almost the same.

Keywords— Movie recommender system, cosine similarity, KNN algorithms, content-based filtering, collaborative filtering, nearest neighbours

1. INTRODUCTION

Recommendation systems are predicting systems that radically recommend items to users or users to the things, and sometimes users to users too. Tech giants like YouTube, Amazon Prime, Netflix use similar methods to recommend video content per their desired interest. because the internet contains huge countless data, finding your content is incredibly difficult and may be very time consuming, thus the advice plays a vital role in minimizing our effort. These systems are becoming more popular nowadays in various areas like in books, videos, music, movies, and other social network sites where the advice is employed to strain the data. it's a tool that's using the user's information to enhance the suggestion result and provides out the foremost preferred choice. User/Customer satisfaction is essential for building the tool. it's beneficial for both customers and firms, because the more satisfied the customer is, the more likely he/she would want to use the system for his or her ease, which might ultimately make revenues for the businesses.

Recommendation system should be improved because the user choice can differ from other users and if the user isn't proud of the result, he/she may not use it again which is that the case with our system.

CHAPTER TWO

2. RELATED WORK

There are many ways of recommending movies using Content-based, Collaborative (User-item, User-user), context based, hybrid methods, and nowadays deep learning is also used to solve this problem. In [1], C. S. M. Wu, D. Garg, and U. Bhandary proposed a recommendation system using collaborative filtering where a user's rating is used to suggest the list. The authors have used the Apache Mahout framework and essentially compared the performances and efficiency of us er-based & item-based recommendations.

In [2],R. E. Nakhli, H. Moradi, and M. A. Sadeghi proposed the percentage view approach for recommending movies to the users, it finds relevant movies for the customer and then compares the performance with a random movie recommendation system for showing the accuracy of the project.

n [3], a content-based recommendation system is proposed by H. W. Chen, Y. L. Wu, M. K. Hor, and C. Y. Tang using neural networks. In recent years, these are top topics for the researchers to work on when they want to build a movie recommendation system.

3. PROBLEM STATEMENT

This recommendation system recommends different movies to users. Since this system is based on a collaborative approach [5], it will give progressively explicit outcomes contrasted with different systems that are based on the content-based approach. Content-based recommendation systems are constrained to people, these systems don't prescribe things out of the box. These systems work on individual users' ratings, hence limiting your choice to explore more. While our system which is based on a collaborative approach computes the connection between different clients and relying upon their ratings, prescribes movies to others who have similar tastes, subsequently allowing users to explore more [6].

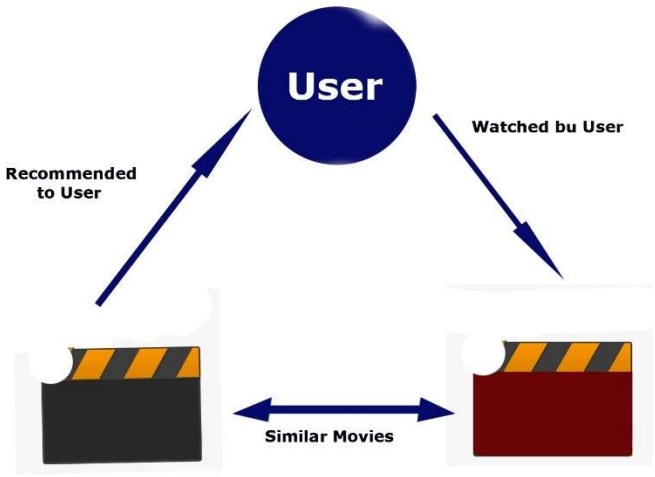
CHAPTER THREE

3. PROPOSED RECOMMENDATIO SYSTEM

The proposed recommendation system used the collaborative filtering technique (item-based approach) which is far more accurate and more efficient to use, as the item-based method can be done offline and because of its non-dynamic nature whereas the user-based changes. The proposed approach uses the KNN algorithm to find the distance between the target movies with every other movie in the dataset and then it ranks the top k nearest similar movies using cosine angle similarity. Different techniques used in this proposed algorithm are discussed below:

3.1 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. Of course, 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:

3.2 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:

$$\boldsymbol{u} \cdot \boldsymbol{v} = [u_1\ u_2\ \ldots\ u_n] \cdot \begin{bmatrix} v_1 \\ v_2 \\ \vdots \\ v_n \end{bmatrix} = u_1 v_1 + u_2 v_2 + \ldots + u_n v_n = \sum_{i=1}^{n} u_i v_i$$

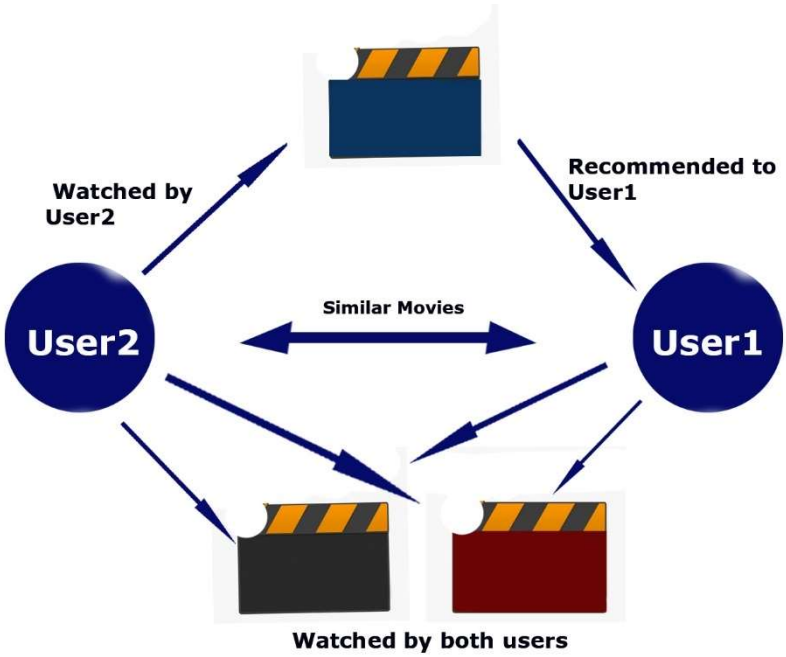
This dot product is important when defining the similarity as it is directly connected to it. The definition of similarity between two vectors u and v is in fact the ratio between their dot products and product of their magnitudes

$$similarity = cos(\theta) = \frac{\boldsymbol{u} \cdot \boldsymbol{v}}{\|\boldsymbol{u}\| \|\boldsymbol{v}\|} = \frac{\sum_{i=1}^{n} u_i v_i}{\sqrt{\sum_{i=1}^{n} u_i^2} \sqrt{\sum_{i=1}^{n} v_i^2}}$$

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

3.3 Collaborative Filtering Recommendation System:

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 hasn't watched it yet, based on its similar user rates.



Algorithms Used:

3.4 K Nearest Neighbours:

The standard method of Collaborative Filtering is known as Nearest Neighborhood algorithm. We have an n × m matrix of ratings, with user u_i, i = 1, ...n and item p_j, j=1, …m. Now we want to predict the rating r_{ij} 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.

$$r_{ij} = \frac{\sum_k Similaries(u_i, u_k) r_{kj}}{number\ of\ ratings}$$

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 user's average ratings of all the items when computing weighted average, and add it back for the target user as shown:

$$r_{ij} = \bar{r}_i + \frac{\sum_k Similaries(u_i, u_k)(r_{kj} - \bar{r}_k)}{number\ of\ ratings}$$

CHAPTER FOUR

4. CONCLUSION

Recommendation systems have become an important part of everyone's lives. With the enormous number of movies releasing worldwide every year, people often miss out on some amazing work of arts due to the lack of correct suggestion. Putting machine learning based Recommendation systems into work is thus very important to get the right recommendations. Similarly, such systems can be improved further by applying neural network embeddings to uplift the quality of recommendations and make them more user personalized. Thus, we conclude that studying various approaches towards recommendation engine is vital to come up with a hybrid engine that overcomes the shortcomings of these independent approaches and multiplies their benefits. Where independent approaches towards a movie recommendation system may have shortcomings, when combined the right way they will help users get the accurate recommendations for movies.

CHAPTER FIVE

REFERENCES

1. C. S. M. Wu, D. Garg, and U. Bhandary, “Movie Recommendation System Using Collaborative Filtering,” In 2018 IEEE 9th International Conference on Software Engineering and Service Science (ICSESS), pp. 11-15, IEEE, 2018 Nov.

2. R. E. Nakhli, H. Moradi, and M. A. Sadeghi, “Movie Recommender System Based on Percentage of View,” In 2019 5th Conference on Knowledge Based Engineering and Innovation (KBEI), pp. 656-660, IEEE.

3. H. W. Chen, Y. L. Wu, M. K. Hor, and C. Y. T ang, “Fully content based movie recommender system with feature extraction using neural network,” In 2017 International Conference on Machine Learning and Cybernetics (ICMLC) , vol. 2, pp. 504-509, Jul, 2017, IEEE.

4. P. Phorasim and L. Yu, “Movie’s recommendation system using collaborative filtering and k-means,” International Journal of Advanced Computer Research, vol. 7, no. 29, p.52, 2017.

5. Su, X., & Khoshgoftaar, T. M. (2009). A survey of collaborative filtering techniques. Advances in artificial intelligence. https://doi.org/10.1155/2009/421425

6. Campadelli, P., Casiraghi, E., & Ceruti, C. (2015, September). Neighborhood selection for dimensionality reduction. International conference on image analysis and processing (pp. 183-191). Cham: Springer.