NITTE (Deemed to be University) | NMAM INSTITUTE OF TECHNOLOGY

Nitte (DU) established under Section 3 of UGC Act 1956 | Accredited with 'A+' Grade by NAAC

Department of Computer Science and Engineering

Report on Mini Project

MOVIE RECOMMENDATION SYSTEM

Course Code: 20CSE71

Course: Artificial Intelligence

Semester: V SEM　　Section: B & C

Submitted To:

Shwetha G K
Assistant Professor
Department of Computer Science and Engineering

Submitted By:

Harshitha J - 4NM20CS077
Pratheeksha - 4NM20CS137

Date of submission:

14 DEC 2022

Signature of Course Instructor

ABSTRACT

Recommender System is a tool helping users find content and overcome information overload. It predicts interests of users and makes recommendation according to the interest model of users. The original content-based recommender system is the continuation and development of collaborative filtering, which doesn’t need the user’s evaluation for items. Instead, the similarity is calculated based on the information of items that are chose by users, and then make the recommendation accordingly. With the improvement of machine learning, current content-based recommender system can build profile for users and products respectively. Building or updating the profile according to the analysis of items that are bought or visited by users. The system can compare the user and the profile of items and then recommend the most similar products. So, this recommender method that compare user and product directly cannot be brought into collaborative filtering model. The foundation of content-based algorithm which is Cosine similarity is used as a metric in different machine

learning algorithms like the KNN for determining the distance between the neighbors, in recommendation systems, it is used to recommend movies with the same similarities and for textual data, it is used to find similarity of texts in the document. As the research of acquisition and filtering of text information are mature, many current content-based recommender systems make recommendation according to the analysis of text information.

This project introduces content-based and popularity-based recommender system for the movies. There are a lot of features extracted from the movie, they are diversity and unique, which is also the difference from other recommender systems. We use these features to construct movie model and calculate similarity. We introduce a new approach for setting weight of features, which improves the representative of movies.

TABLE OF CONTENTS

INTRODUCTION

In this age of the internet, the quantity of data transactions that happen every minute has increased exponentially. The huge amount of data has dramatically increased with the number of users on the internet. However, not all the data available on the internet is of use or provides satisfactory results to the users. Data in such huge volumes often turns out to be inconsistent and without proper processing of this information, it gets wasted. In such cases, users have to run their search multiple times before they finally obtain what they were originally looking for. To solve this problem, researchers have come up with recommendation systems. A recommendation system provides relevant information to the users by taking into account their past preferences. Data is filtered and personally customized as per the user requirements. With more and more data available on the Internet, recommendation system

have become really popular, due to their effectiveness in providing information in a short time-span. Recommender systems have been developed in various areas such as music, movies, news, and products in general. In today’s age, a majority of organizations implement recommendation systems for fulfilling customer requirements. LinkedIn, Amazon, and Netflix are just a few to name. LinkedIn recommends relevant connections of the people the user might know among the millions that are subscribed on the portal.

This way, the user does not have to run extensive searches for people manually. Amazon recommendation systems work such that they suggest correlated items that the customers can purchase. If a certain customer prefers buying books from the shopping portal, Amazon provides suggestions related to any new arrivals in previously preferred categories. In a very similar way, Netflix takes into account the types of shows that a customer watches, and provides recommendations similar to those. By the method in which recommendation systems work, they can be broadly classified into three categories—Content-based, Collaborative and Hybrid approach. A content-based recommendation system considers the user’s past behavior and identifies patterns in them to recommend to recommend items that are similar to them. Collaborative filtering analyses the user’s previous experiences and ratings and correlates it with other users. Based on the ones that have the most similarity, recommendations are made.

PROBLEM STATEMENT

The goals of this thesis project is to do the research of Recommender Systems and find a suitable way to implement. There are many kinds of Recommender Systems but not all of them are suitable for one specific problem and situation. Our goal is to find a new way to improve the classification of movies, which is the requirement of improving content-based recommender systems

OBJECTIVES

For building a recommender system from scratch, we face several different problems. Currently there are a lot of recommender systems based on the user information, so what should we do if the website has not gotten enough users. After that, we will solve the representation of a movie, which is how a system can understand a movie. That is the precondition for comparing similarity between two movies. Movie features such as genre, actor and director are a way that can categorize movies. But for each feature of the movie, there should be different weight for them and each of them plays a different role for recommendation. So, we get these questions:

- How to recommend movies when there is no user information.
- What kind of movie features can be used for the recommender system.
- How to calculate the similarity between two movies.
- Is it possible to set weight for each feature

Required Libraries

Pandas and numpy:

importing these two important libraries for data analysis and manipulation

CountVectorizer:

used to convert a collection of text documents to a vector of term/token counts.

cosine_similarity:

metric of our similarity matrix

IMPLEMENTATION AND METHODOLOGY

The approach used for building the recommendation system is content-based filtering. As discussed earlier, content-based filtering analyses user's past behavior and recommends items similar to it based on the parameters considered. This aims at recommending movies to users based on similarity of genres. If a user has rated high for a certain movie, other movies containing similar genres are recommended by the system. The dataset used in for this purpose is subdivided into two sections.

One section contains the list of movies along with the genres that they have been categorized under. The other part of the dataset contains a list of ratings of movies that have been rated by the user on a scale of 1–5, with 5 being the highest. First a combined dataset of movies, genres and their ratings have to be constructed for correlating genres with the ratings. For the sake of simplicity, the ratings have been converted to binary values. If the rating given by a particular user is greater than3, it receives a value of 1, otherwise it receives a value of

−1. The genres are also segregated in a binary format, maintaining a consistent approach. Out of the set of11 genres that are present in total, if a movie has a certain genre, it receives the value of 1. If the genre is not present in the movie, it receives a value of 0. The user profile matrix provides a combined effect of the genres and ratings by computing the dot product of the genre and the ratings matrix. Again, for the sake of consistency, a binary format is adopted. If the dot product is a negative value, 0 is assigned to it. For a positive value, 1 is assigned to it. After obtaining a dot product matrix of all the movies, a similarity measure is calculated by computing the least distance between the user under consideration and the others. The values which have the least deviation with respect to the current user's preferences are the ones that are recommended by the system. The algorithm adopted for building the recommendation system is given below:

Algorithm Step 1. Construct a data frame of the genre dataset with movie ID as the rows and genres as columns separated by pipeline character.

Step 2. Make a list of all the genres that are available in the dataset.

Step 3. Iterate through the previously made genre data frame. If a genre is presenting a movie, value of 1 is assigned to the genre matrix.

Step 4. Read the ratings sheet and construct a ratings matrix which assigns value for movies

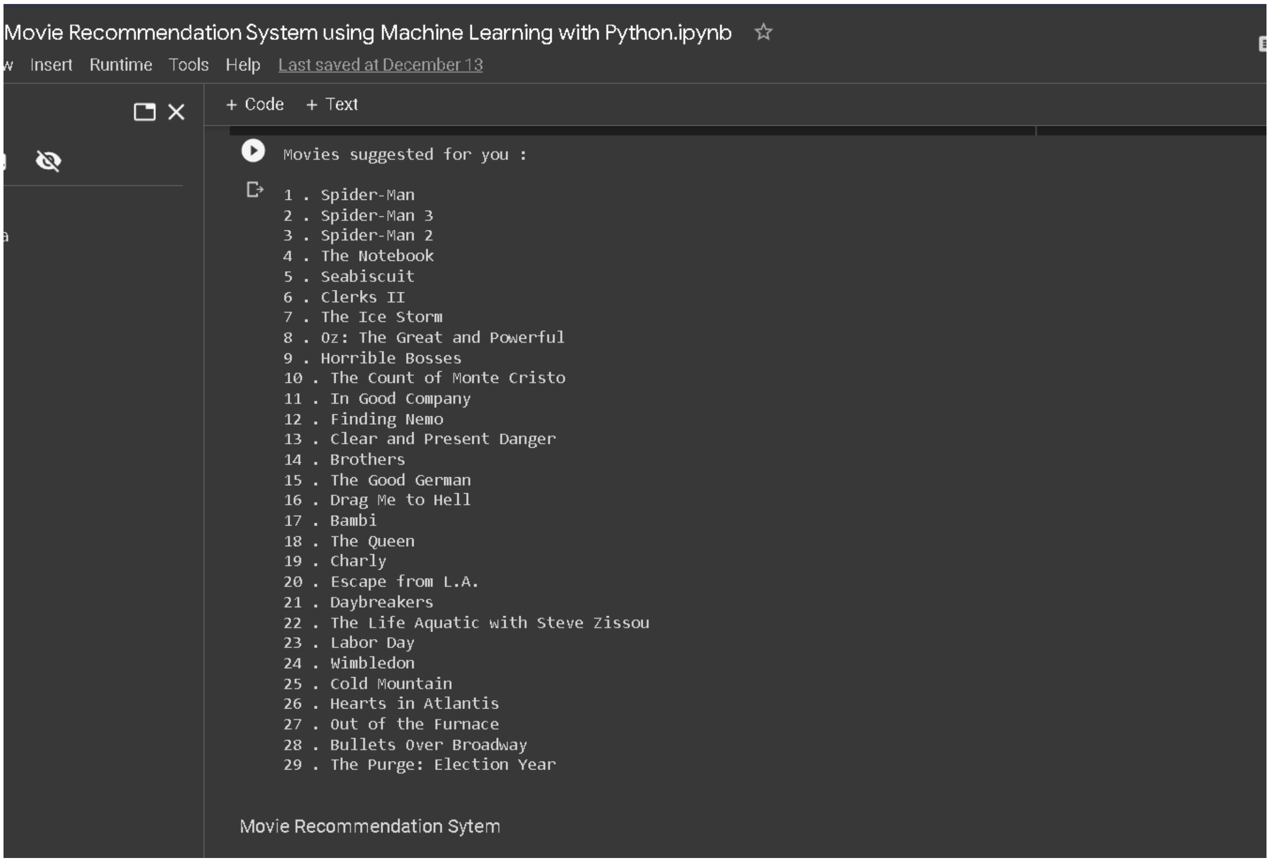
Step 5. Calculate the dot product of the two matrices—genre matrix and ratings matrix. This is the result matrix

Step 6. Convert the result matrix to a binary format. For a negative dot product value, assign 0, else assign a value of 1.

Step 7. Calculate the Euclidian distance between the current user and other users.

Step 8. Retain the rows which have the minimum distance. These are the recommended movies for the current user

RESULTS AND DISCUSSIONS


Movie Recommendation System using Machine Learning with Python.ipynb
w Insert Runtime Tools Help Last saved at December 13
+ Code + Text
Movies suggested for you :
1 . Spider-Man
2 . Spider-Man 3
3 . Spider-Man 2
4 . The Notebook
5 . Seabiscuit
6 . Clerks II
7 . The Ice Storm
8 . Oz: The Great and Powerful
9 . Horrible Bosses
10 . The Count of Monte Cristo
11 . In Good Company
12 . Finding Nemo
13 . Clear and Present Danger
14 . Brothers
15 . The Good German
16 . Drag Me to Hell
17 . Bambi
18 . The Queen
19 . Charly
20 . Escape from L.A.
21 . Daybreakers
22 . The Life Aquatic with Steve Zissou
23 . Labor Day
24 . Wimbledon
25 . Cold Mountain
26 . Hearts in Atlantis
27 . Out of the Furnace
28 . Bullets Over Broadway
29 . The Purge: Election Year
Movie Recommendation Sytem

CONCLUSION AND FUTURE SCOPE

We have illustrated the modelling of a movie recommendation system by making the use of content-based filtering in the movie recommendation system. The KNN algorithm is implemented in this model along with the principle of cosine similarity as it gives more accuracy than the other distance metrics and the complexity is comparatively low too Recommendations systems have become the most essential fount of a relevant and reliable source of information in the world of internet. Simple ones consider one or a few parameters while the more complex ones make use of more parameters to filter the results and make it more user friendly. With the inclusion of advanced deep learning and other filtering techniques like collaborative filtering and hybrid filtering a strong movie recommendation system can be built. This can be a major step towards the further development of this model as it will not only become more efficient to use but also increase the business value even further.

REFERENCES

1.https://towardsdatascience.com/using-cosine-similarity-to-build-a-movie-recommendation-system-ae7f20842599

2.https://www.sciencedirect.com/science/article/pii/S2666285X22000176