MOVIE RECOMMENDATION SYSTEM

Seminar (IT290) Report

Submitted in partial fulfilment of the requirements for the degree of

BACHELOR OF TECHNOLOGY

In

INFORMATION TECHNOLOGY

by

MAYUR JINDE (2010759)

DEPARTMENT OF INFORMATION TECHNOLOGY

NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA

SURATHKAL, MANGALORE - 575025

MAY, 2022

D E C L A R A T I O N

I hereby *declare* that the *Seminar (IT290) Report* entitled "**Movie Recommendation System**" which is being submitted to the National Institute of Technology Karnataka Surathkal, in partial fulfilment of the requirements for the award of the Degree of Bachelor of Technology in the department of Information Technology, is a ***bonafide report of the work carried out by me.*** The material contained in this seminar report has not been submitted to any University or Institution for the award of any degree.

Mayur Jinde – 2010759

_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

Signature of the Student

Place : NITK, Surathkal
Date : 3rd May, 2022

CERTIFICATE

This is to certify that the Seminar entitled **"Movie Recommendation System"** has been presented by ***Mayur Jinde - 2010759***, a student of IV semester B.Tech. (IT), Department of Information Technology, National Institute of Technology Karnataka, Surathkal, on 30th April, during the even semester of the academic year 2021 - 2022, in partial fulfillment of the requirements for the award of the degree of Bachelor of Technology in Information Technology.

_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

Guide Name

_ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _ _

Signature of the Guide with Date

Place : NITK, Surathkal
Date : 3rd May, 2022

Abstract - Multimedia is considered as one of the best sources of entertainment. People of all age groups love to watch movies. Movie Recommender System is essential in our social lives as it enhances the field of entertainment. The proposed system on Movie Recommendation System caters the requirements of the user. The major aim is to provide crisp relevant content to the end-users out of semi-structured content on the internet. The main purpose is to generate accurate, efficient and personalized recommendations to the user. The recommender system is generated by applying Cosine Similarity Algorithm. As a result, the live working of the system generates accurate and personalized recommendations along with the analysis of sentiments for the end users. It is also concluded that Cosine Similarity provides better and efficient results for a recommender system. Over the past years, the internet has broadened the horizon of various domains to interact and share meaningful information. As it is said that everything has its pros and cons therefore, along with the expansion of domain comes information overload and difficulty in extraction of data. To overcome this problem the recommendation system plays a vital role. It is used to enhance the user experience by giving fast and coherent suggestions. This paper describes an approach which offers generalized recommendations to every user, based on movie popularity and/or genre. Content-Based Recommender System is implemented using various deep learning approaches. This paper also gives an insight into problems which are faced in content-based recommendation system and we have made an effort to rectify them.

Keywords: Recommendation System, Content-Based Recommender System, Deep learning.

Table of Contents Page no.

List of Figures

Chapter – 1 INTRODUCTION

Advancement in technology is reaching new heights every day and due to which we can see enormous growth in information. To deal with such large data we use machine learning that automates analytical model building. The early classification of machine learning is divided into three broad categories: Supervised learning, Unsupervised learning and Reinforcement learning. We use computers to make predictions to help us achieve better results using various computational statistics. Tasks can be performed without being explicitly programmed to do so. It becomes a tedious task to extract the relevant information. Search engines solve the problem to some extent but it does not solve the personalization problem. Recommendation System framework plays a vital role in today's internet surfing, be it buying a product from an e-commerce site or watching a movie on some video-on-demand service. In our everyday life, we depend on recommendations given by other people either by word of mouth or reviews of general surveys. People often use recommender systems over the web to make decisions for the items related to their choice. Recommendation systems are software tools and techniques whose goal is to make useful and sensible recommendations to a collection of users for items or products that might interest them. In other words, the recommender system or recommendation systems belongs to a class of information filtering system that aims at predicting the 'preference' or 'rating' given to an item.

With the increase of World Wide Web and high-speed internet, multimedia is turned out as one of the best sources of entertainment. People are keenly interested in watching movies. However, there are thousands of movies available and it becomes extremely difficult for people to choose a suitable movie. So, Movie Recommendation Systems solves this problem by providing accurate suggestions based on people interests. It also recommends movies that people are generally not aware of and thus saves a lot of time. For ex: A person who is a fan of Avengers Series and has already watched "The Avengers", "Avengers: Infinity War" and "Avengers: Age of Ultron". So, the recommendation system would suggest "Avengers: The End Game.

Recommendation systems are primarily using three approaches. In content-based filtering, we do profiling based on what type of content any user is interested in and using the collected information,

it recommends items. It basically recommends those items to the user that seems to fulfils his/her needs and also analyses the sentiments on the reviews given by the user for that movie. Movie recommendation systems provide a user with the movie suggestions that are more likely to be watched by him using some means like, the user's past behavior, or user's profile etc. It generates personalized recommendations based on user's likes, ratings and dislikes. Another one is collaborative filtering, where we make clusters of similar users and use that information to make recommendations. Hybrid systems are the one which takes into account both above stated approaches to deal with operational data more concisely. Our goal is to provide accurate recommendations with less computational complexity.

The aim is to deliver a Movie Recommendation System which caters all the requirements of the user. Due to high-speed internet and various platforms releasing movies of different genres, people of different generations binge watch the latest movies and shows. All these addicted users desire to watch movies of their interests. Movie Recommendation System caters to all the needs of the users such that the user feels comfortable and is able to interact with the system. So that's what the proposed system does it recommends the movie according to user's interest.

Nowadays people are more active on social media, online shopping websites, Netflix, amazon prime and so on. People also share their thought about a particular product by writing the reviews or comments about it. While online shopping or online watching movies people generally go through the reviews of the item which is the main source of someone's sentiments. Sentiment analysis helps in making business decision. Sentiment analysis is used to highlight the week and good point of the product so that one can make better business decision. As we have discussed earlier user depends on other to suggest some content of their personal choice and interest which they can binge watch in their free time. So, the authors are building this system to remove this kind of dependency on others because it is difficult for both the sides to involve in such an activity which wastes time and result will also not be that much accurate as it is given by a recommender system. Hence, with the help of user's preferences and interest a better structure with accurate recommendations is provided to the users.

Chapter – 2 LITERATURE REVIEW

Recommender systems can be utilized in many contexts to generate recommendations to the users that might interest them. Recommendation systems were first developed by Tapestry project in 1992[5]. The Tapestry project (first commercial recommender system) introduced the term "collaborative filtering" Existing movie recommendation systems are mostly built using the content based and collaborative filtering approach. They show the recommended movies to the user on the basis of user's priorities using collaborative filtering. It is a technique that is used to filter those items which a user might like on the ground of reaction by similar users [6]. It works by finding smaller group of users from the larger set of people. While content-based Filtering i.e., another approach uses an item feature in order to recommend other items with similar features or properties. Except these two approaches there are also some other approaches proposed in the past which is discussed below. Harsh Khatter, Anil Kumar Ahlawat [1] have proposed a model based on content curation algorithms which are based on personalized web searching. They suggested a new technique which is to use clustering technique for recommendations along with association rule mining. The system proposed by them automatically provides the recommendation to the user as soon as he/she logged in. The expected outcome of this model is the top N relevant recommendations to the user based on the user's interest. M. Chenna Keshava,S. Srinivasulu, P. Narendra Reddy and B. Dinesh Naik[2] focused on providing recommendations to the user based on the user's ancient data. The authors also committed to provide accurate recommendations and suggest much more content to an individual. They have used the concept of CineMatch. They increased CineMatch Algorithms by about 10% with the use of some Collaborative Filtering Techniques. The authors have used various ML algorithms like XGBoost for featuring the data, Surprise Baselineonly Model for training and testing data, Surprise KNN Baseline Model for updating the data by using the features obtained from Surprise Baselineonly Model, Matrix Factorization SVD for using next to update the data at every particular instance of time, Matrix Factorization SVDpp for updating the model with the obtained features. They have concluded that the best model was Matrix Factorization SVDpp which provided the RMSE Value of 1.0675. V.R.Azhaguramyaa, Hemanshu P Thakker, Murali Manohar K S, and Mithun K[3] have discussed the RNN algorithm for product recommendation. The authors have used Recommendation system as a filtering system which is used to predict the product that the user would like to brought or purchase. The proposed system recommends the movie according to the Comments and ratings provided by the user.

Chapter – 3 TECHNICAL DISCUSSION

3.1 Methodology

The movies are recommended based on a simple algorithm called **Cosine Similarity**. Cosine similarity is a measure used to determine the similarity between two items. Mathematically it can be determined as the cosine angle between two vectors in a three-dimensional plane. We can also check the Euclidean distance between the two vectors to determine how different or similar they are from each other. In our case, one of the vectors is the movie that is searched and the rest of the movies in the database are checked as the second vector. The top ten movies which have the least Euclidean distance corresponding to the searched movie are shown as recommendations.

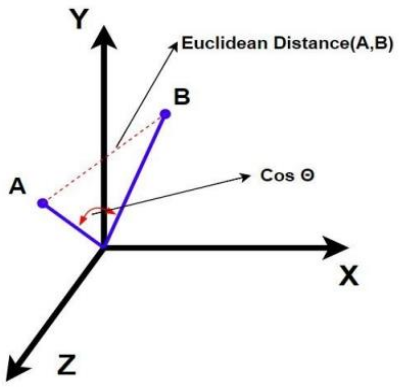

Fig 3.1 Cosine Similarity

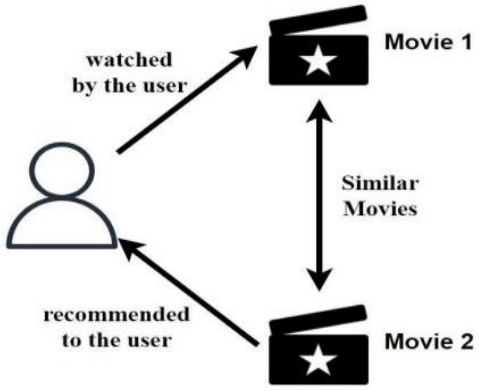

Fig 3.2 Content Based Filtering

Cosine Similarity is a type of Content-based filtering approach. It is one of the most popular techniques used in recommendation systems. The attributes of a thing are termed as" content". Based on these attributes we are able to classify whether the two things are similar or not. The attributes can be words specified in the database such as genre, cast names, director names, description, and so on. If the attributes match or have a high similarity then the two movies can be classified as similar movies. The intuition behind this sort of recommendation system is that if a user liked a particular movie or show, he/she might like a movie or a show similar to it.

3.1.1 Working

The Implementation of this project is done using Google Colab which involves various steps as follows:

Step – 1: **Data collection** - The movies dataset is collected from Kaggle and Wikipedia. The data from Kaggle is for movies up till year 2016 and dataset for movies after 2016 was taken from Wikipedia. Then, both datasets are combined in a common format to get the final dataset in CSV file. This file contains various details about the like the movie name, cast, director, genre etc. Approximately 4800 movies are present in the CSV file with 25 attributes each.

Step – 2: **Data Preprocessing** - Processing and filtering the missing data from the CSV file.

Step – 3: Feature Extraction - The original dataset from Kaggle consisted 25 features in total including movie title, director name, release date, revenue, runtime, language, etc. A lot of the features are irrelevant for the proposed model and do not really contribute to the end result. Dataset was finally reduced down to following 5 columns – genres, keywords, tagline, cast, director.

Another column named "combined_info" was added in the final dataset. Here, combine_info feature is the space separated collection of other features except the movie_title which will be used for creating the similarity matrix.

Step – 4: **User input** - User to input his/her favorite movie name.

Step – 5: Applying Cosine Similarity - Cosine similarity calculates the cosine angle between two vectors as shown in Fig 3.1, which represents the similarity of those two vectors. The lower the cosine of two vectors, the more similar they are.

Cosine similarity is basically the dot product of two vectors divided by the magnitudes of two vectors.

$$similarity = \cos(\theta) = \frac{A.B}{|A||B|}$$

- A.B = dot product of the vectors 'x' and 'y'.
- ||x|| and ||y|| = length of the two vectors 'x' and 'y' respectively.

It ranges from 0 to 1 with 0 being the lowest (the least similar) and 1 being highest (the most similar). Cosine similarity matrix is created using cosine_similarity() function present in python's "sklearn.metrics.pairwise" package. The scores are calculated over the combined_info feature of our dataset which includes genres, keywords, tagline, cast, director of the movies.

Step – 6: **Movie Recommendation** - When the system has to recommend movies similar to the given movie, it reads the cosine similarity matrix row for that movie and checks the similarity score of this movie with other movies. Then, the first 30 movies having higher score are recommended to the user.

3.1.2 Comparative Analysis

Here the proposed algorithm is compared with various other existing algorithms. There are many similarity metrics available like Jaccard coefficient, dice coefficient, correlation based, Euclidean distance, Pearson correlation coefficient. Each of them has some advantages and limitations. The authors found cosine similarity most suitable for the proposed system based on the following studies –

- In the survey, an experiment comparing different similarity metrics was found which concludes that "Cosine and extended Jaccard similarity takes less execution time as compared to the adjusted based similarity and correlation-based similarity".
- However, it was observed that when the users are less cosine similarity behaves better but as the number of users increases, the extended Jaccard similarity behaves much better.
- In another study, multiple similarity measures were compared on a books rating dataset.
- Namely, Pearson correlation, Euclidean distance, Cosine similarity and Jaccard coefficient were used in the study. Pearson correlation, Euclidean distance and Cosine similarity algorithms consider only the common items that have been rated for measuring the similarity, whereas Jaccard coefficient considers the common items as well as the items that are present in either of the entity.
- The study states that "Jaccard coefficient is not a good choice to opt when we want to consider only the common item ratings".
- Finally, the authors chose to consider the better performance of cosine similarity over better computing time of Jaccard coefficient.

3.2 Experimental Results

When the user runs the program, he is first asked to enter his/her favorite movie name, and then the system shows top 30 recommended movies similar to the input movie by the user with the most similar on the top. Even if the user makes any spelling mistake in typing the movie name the system gives appropriate results for most of the time.

Click here to view the source code of my project.

The results for a few movies are shown below –

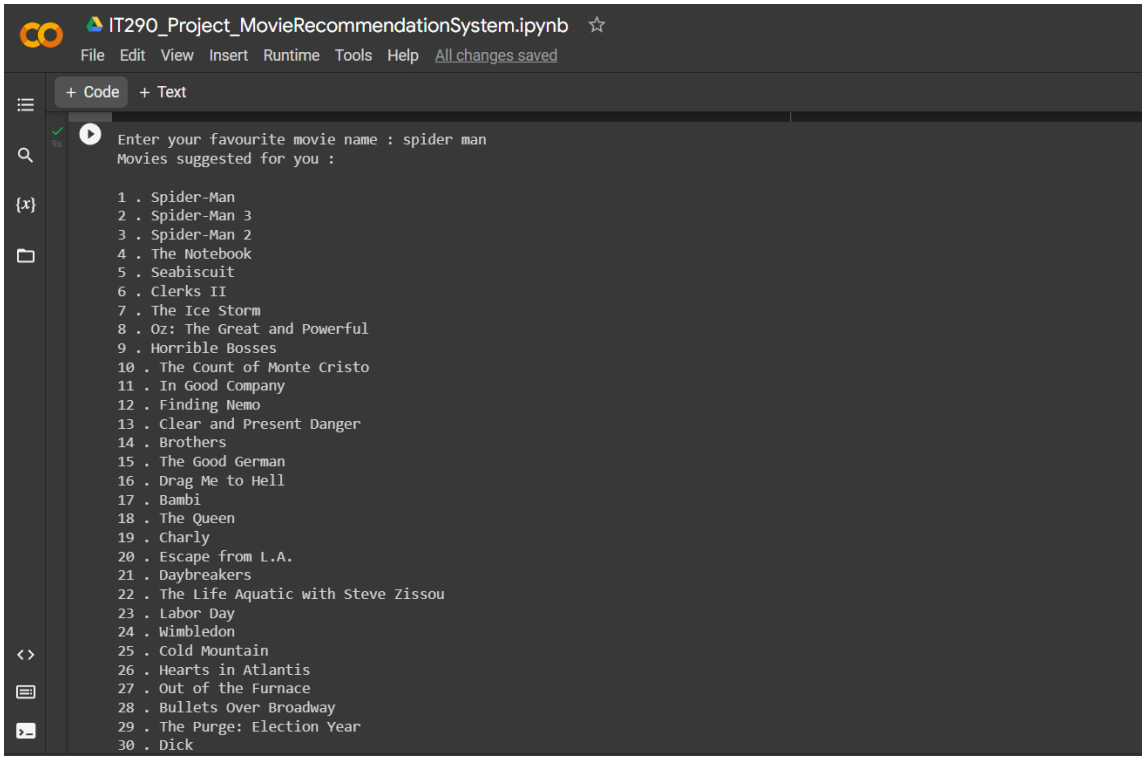

Fig 3.2.1 Result-1

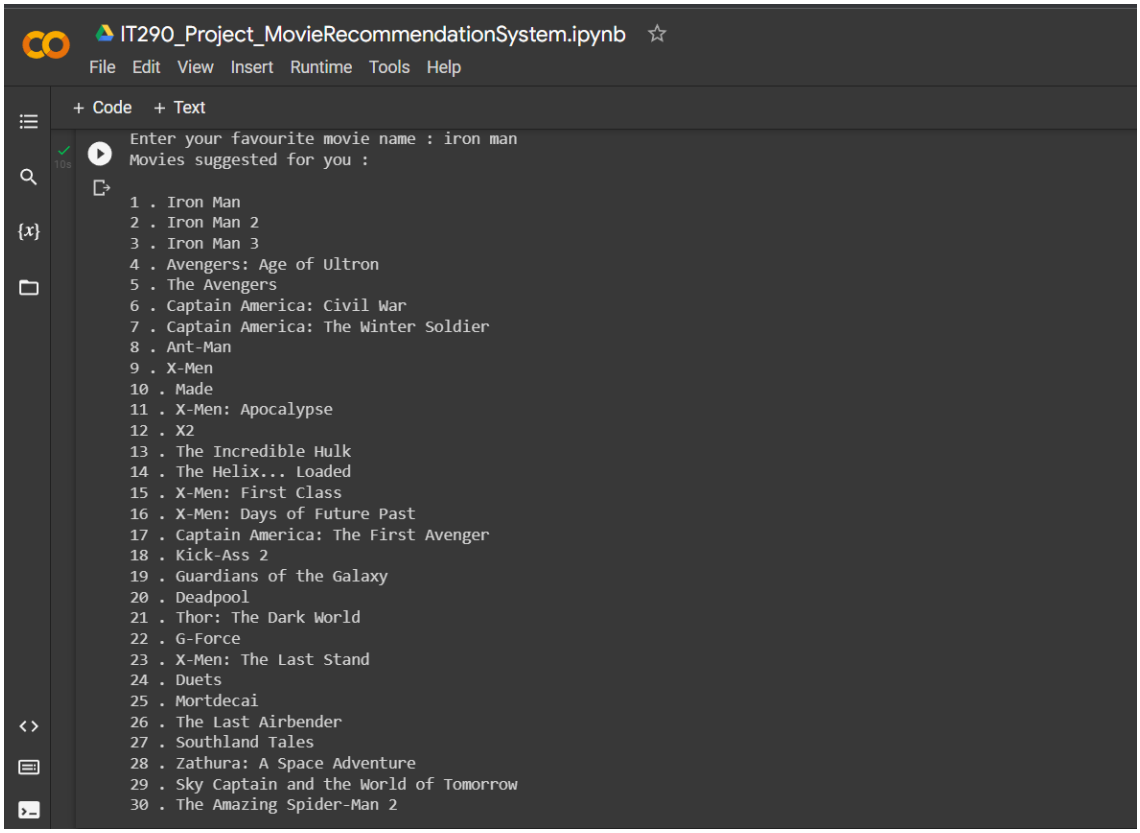

Fig 3.2.2 Result-2

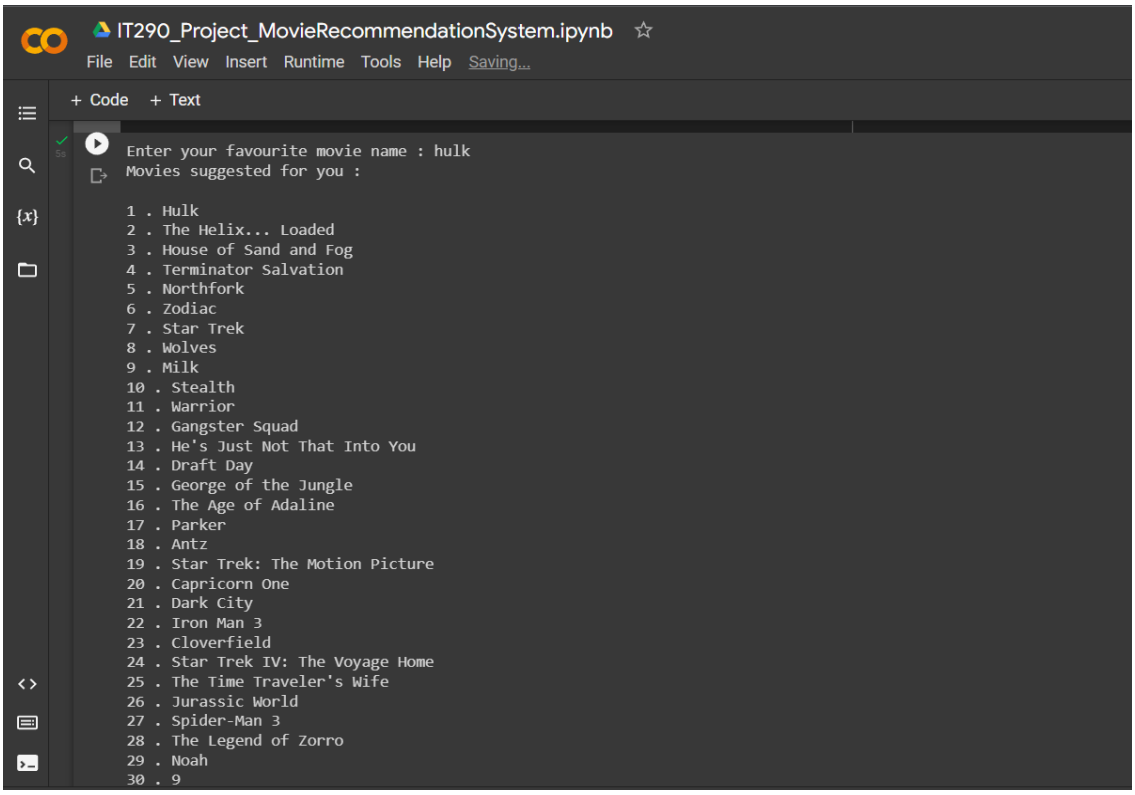

Fig 3.2.3 Result-3

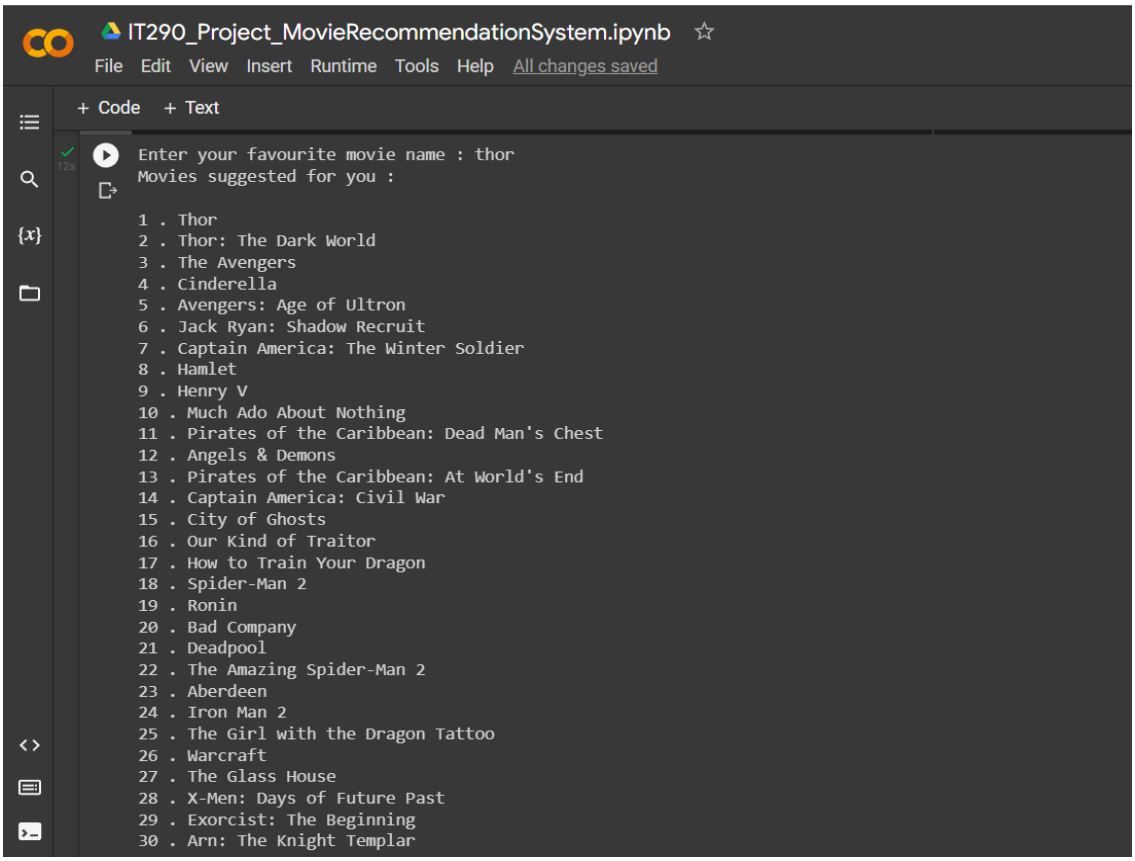

Fig 3.2.4 Result-4

Chapter – 4 CONCLUSIONS AND FUTURE TRENDS

We have illustrated the modelling of a movie recommendation system by making the use of content-based filtering in the movie recommendation system. The principle of Cosine Similarity is implemented in this model 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.

Cosine similarity is used over many other available similarities for the recommendation system because it has better computing time and efficiency than others. Another advantage of cosine similarity is that it can still give smaller angle between two similar objects even if they are far apart by the Euclidean distance. This approach can be used as a base for other recommender systems which recommends songs, books, news, videos etc. It can also be incorporated into various ecommerce websites. Also, the system can further be improved such that the user can also provide rating and comment on the movies.

5 REFERENCES

[1] Khatter, Harsh and Kumar Ahlawat, Anil, "Analysis of Content Curation Algorithms on Personalized Web Searching" (March 29, 2020). Proceedings of the International Conference on Innovative Computing & Communications (ICICC) 2020.

[2] M. Chenna Keshava, S. Srinivasulu, P. Narendra Reddy, B. Dinesh Naik, "Machine Learning Model for Movie Recommendation System," International Journal of Engineering Research & Technology (IJERT), ISSN: 2278-0181 IJERTV9IS040741 Vol. 9 Issue 04, April-2020, DOI: 10.17577/IJERTV9IS040741.

[3] Prof.V.R.Azhaguramyaa, Hemanshu P Thakker, Murali Manohar K S, Mithun K, "Smart Product Recommender System using Machine Learning," International Journal of Advanced Science and Technology ,Vol. 29, No. 9s, (2020).

[4] Akansh Surendran, Aditya Kumar Yadav, Aditya Kumar, "Movie Recommendation System using Machine Learning Algorithms". International Research Journal of Engineering and Technology 2020.

[5] R. Ahuja, A. Solanki and A. Nayyar, "Movie Recommender System Using K-Means Clustering AND K-Nearest Neighbor," 2019 9th International Conference on Cloud Computing, Data Science & Engineering (Confluence), Noida, India, 2019, pp. 263-268, doi: 10.1109/CONFLUENCE.2019.8776969.

[6] Nirav Raval, Vijayshri Khedkar, “Collaborative Filtering Based Movie Recommendation System,” International Journal of Scientific & Technology Research,Vol 8, Issue 12, Dec 2019.