MOVIE RECOMMENDATION SYSTEM

Mini Project KCS 752/ 7th Sem

STUDENT NAME :
Aditya Kumar
Srivastav
[2000320120018]

Outline

- Abstract
- Introduction
- Need and Impact
- Application
- Problem Statement
- Goal
- Literature Review
- Methodology
- Tools and Technology Used
- References

ABSTRACT

- **Recommender systems** encompass a class of techniques and algorithms that can suggest "relevant" items to users. They predict future behavior based on past data through a multitude of techniques including **matrix factorization**. For streaming services like Netflix, recommendation systems are essential for helping users find new movies to enjoy.

- Three main approaches are used for recommender systems. One is **Demographic Filtering** i.e They offer generalized recommendations to every user, based on similar demographic features. Second is **Content-based filtering**, where users interests are profiled 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.

- In this project, we propose a machine learning approach to produce a **Content-based filtering** system which predicts movie recommendations for a user based on large database of continuously updated movies.

Introduction

- In the era of information overload, it is very difficult for users to get information that they are really interested in. And for the content provider, it is also very hard for them to make their content stand out from the crowd. That is why many researchers and companies develop Recommender System to solve the contradiction. The mission of Recommender System is to connect users and information, which in one way helps users to find information valuable to them and in another way push the information to specific users. This is the win-win situation for both customers and content providers.

- Recommendation systems help users find and select items (e.g., books, movies, restaurants) from the huge number available on the web or in other electronic information sources. Given a large set of items and a description of the user's needs, they present to the user a small set of the items that are well suited to the description.

Existing System

- Movie Recommendation: Netflix uses algorithm for recommending movies according to their interest. Other such platforms that provide recommendations include hotstar, sonyLIV, voot, ALTBalaji etc.

- Music Recommendation: e Spotify, JioSavan, Gaana etc.

- News: Google News, Apple News, Flip board, Tweet Deck, Pocket etc.

- Fashion: Myntra, Amazon, Club Factory, SHEIN, Lime Road, Flipcart etc.

Proposed System

- The proposed system is to build a movie recommendation engine which consists of the following steps.
- Perform Exploratory Data Analysis (EDA) on the data
- Build the recommendation system
- Get recommendations

Objectives

- To find a suitable way to implement a system which predicts movie recommendations for a user based on large database of continuously updated movies

- To find a new way to improve the classification of movies, which is the requirement of improving content-based recommender systems.
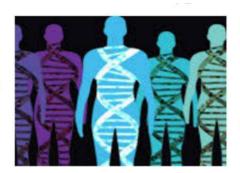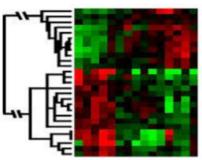
Methodology Used

There are various types of recommender systems with different approaches, some are mentioned below :

- Demographic Filtering
- Content-based Filtering
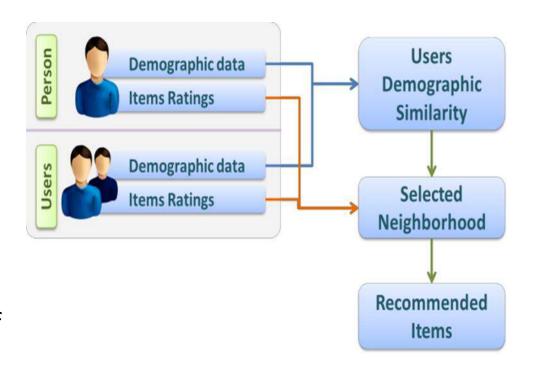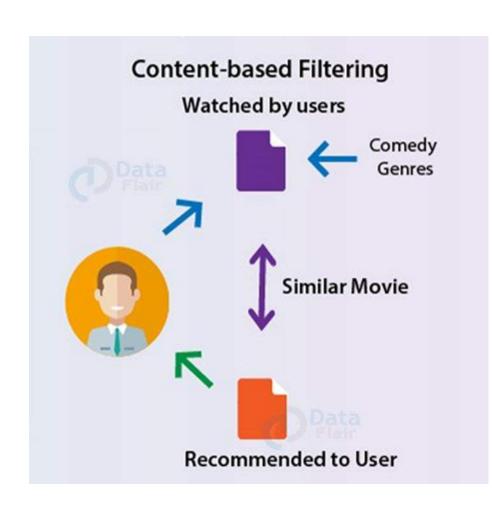- Collaboration-based Filtering

Demographic filtering

They offer generalized recommendations to every user, based on movie popularity with similar demographic features. The basic idea behind this system is that movies that are more popular and critically acclaimed will have a higher probability of being liked by the average audience.
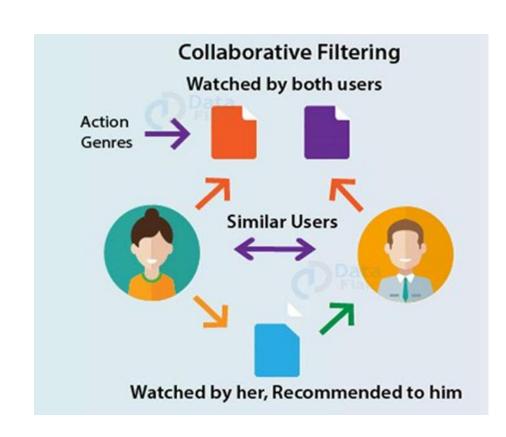
CONTENT BASED FILTERING

- Content-based filtering
- where we try to profile the users interests using information collected, and recommend items based on that profile.
- It is based on similarity index.
- Ex. YouTube.

COLLABORATIVE FILTERING

- Collaborative filtering

- Collaborative filtering approaches build a model from user's past behaviour (i.e. items purchased or searched by the user) as well as similar decisions made by other users. This model is then used to predict items (or ratings for items) that user may have an interest in.
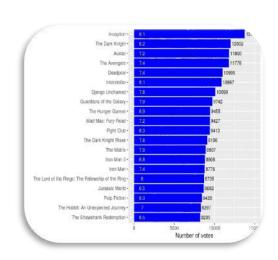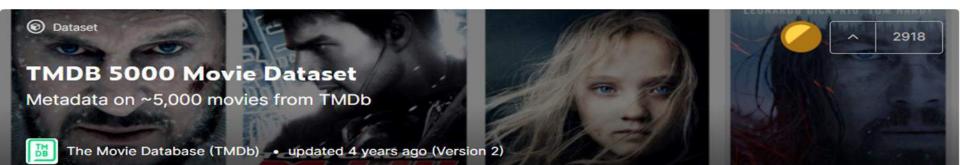
- Example: Facebook

DATASET

▥ tmdb_5000_credits.csv
▥ tmdb_5000_movies.csv

- TMDB 5000 Movie Dataset

- contains metadata for over 45,000 movies.
- tmdb_5000_movies.csv: The Information about the items –
- Budget | genres | homepage| id| keywords | original_language| Production|spoken |runtime| tagline …etc.
- tmdb_5000_credits.csv: crew |cast | movie_id.

© Dataset

TMDB 5000 Movie Dataset
Metadata on ~5,000 movies from TMDb

TMDB The Movie Database (TMDb)   •   updated 4 years ago (Version 2)

2918

Implementation

```
In [62]:  # finally we have our new dataset ready.
          new_df.head()
```

Out[62]:

	movie_id	title	tags
0	19995	Avatar	in the 22nd century, a paraplegic marine is di...
1	285	Pirates of the Caribbean: At World's End	captain barbossa, long believed to be dead, ha...
2	206647	Spectre	a cryptic message from bond's past sends him o...
3	49026	The Dark Knight Rises	following the death of district attorney harve...
4	49529	John Carter	john carter is a war-weary, former military ca...

Implementation

```
In [178...  recommend('Avatar')
```

```
Aliens vs Predator: Requiem
Aliens
Falcon Rising
Independence Day
Titan A.E.
```

```
In [182...  recommend('Captain America: The First Avenger')
```

```
Captain America: Civil War
Red Tails
Saints and Soldiers
Letters from Iwo Jima
Captain America: The Winter Soldier
```

```
In [183...  recommend('Deadpool')
```

```
Ant-Man
Iron Man 2
MacGruber
The Animal
The Amazing Spider-Man
```

```
In [188...  recommend('Inception')
```

```
12 Rounds
RED
Abduction
Krrish
The Animal
```

```
In [ ]:  #finally, we have our recommendation system ready
         #It's time to get recommendation.
```
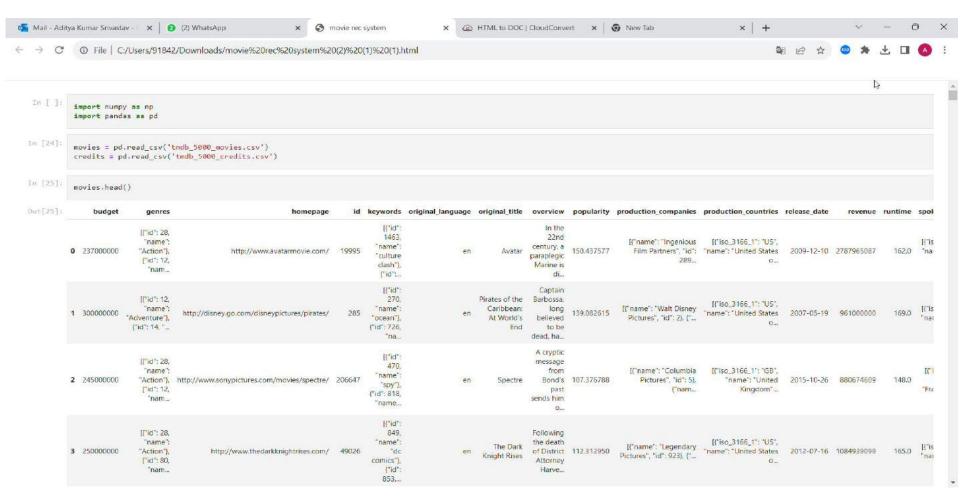
Implementation

- file:///C:/Users/91842/Downloads/movie%20rec %20system%20(2)%20(1)%20(1).html

Results

Tools and Technologies Used

- Python

- Python is an interpreted, high-level, general-purpose programming language Created by Guido van Rossum. Python is dynamically typed and garbage-collected. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming.

- Jupiter Notebook

- The Jupyter Notebook is an open-source web application that allows you to create and share documents that contain live code, equations, visualizations and narrative text. Uses include: data cleaning and transformation, numerical simulation, statistical modeling, data visualization, machine learning, and much more.

LIBRARIES USED

- NumPy: NumPy is a library for the Python programming language, adding support for large, multi-dimensional arrays and matrices, along with a large collection of high-level mathematical functions to operate on these arrays.

Pandas: pandas is a software library written for the Python programming language for data manipulation and analysis.

Scikit-learn: Scikit-learn is a free software machine learning library for the Python programming language. To build our model, we first create a count matrix that is created by the help of a count vectorizer.

References

- Kumar, M., Yadav, D. K., Singh, A., & Gupta, V. K. (2015). A movie recommender system: Movrec. International journal of computer applications, 124(3), 7-11.

- Scharf, P. C., & Alley, M. M. (1993). Spring nitrogen on winter wheat: II. A flexible multicomponent rate recommendation system. Agronomy journal, 85(6), 1186-1192.

- Goldberg, D., Nichols, D., Oki, B. M., & Terry, D. (1992). Using collaborative filtering to weave an information tapestry. Communications of the ACM, 35(12), 61-70.

- Sarwar, B., Karypis, G., Konstan, J., & Riedl, J. (2001, April). Item-based collaborative filtering recommendation algorithms. Proceedings of the 10th international conference on World Wide Web (pp. 285-295). Hong Kong: Association for Computing Machinery.

THANK YOU