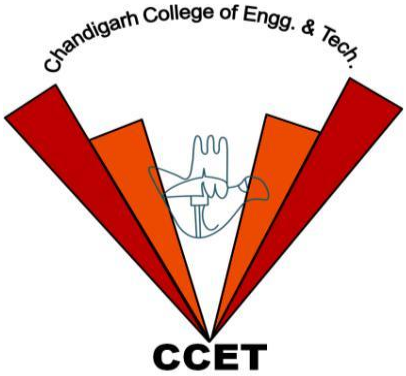
Lab Project Report

On

MOVIE RECOMMENDER SYSTEM USING DATA MINING AND ANALYSIS

Under the supervision of



CHANDIGARH COLLEGE OF ENGINEERING AND TECHNOLOGY (DEGREE WING)

Government Institute under Chandigarh (UT) Administration, Affiliated to Panjab University, Chandigarh

Sector-26, Chandigarh. PIN-160019

JANUARY-JULY, 2021

Submitted by
ABHINAV RAWAT (CO18304)
ABHISHEK PANT (CO18305)
SHUVAM ROY (CO18349)
CSE 6th SEM

MOVIE RECOMMENDER SYSTEM USING DATA MINING AND ANALYSIS

Abstract

This paper discuss about recommendations of the movies. A movie recommendation is important in our social life due to its strength in providing enhanced entertainment. Such a system can suggest a set of movies to users based on their interest, or the popularities of the movies. A recommendation system is used for the purpose of suggesting items to purchase or to see. They direct users towards those items which can meet their needs through cutting down large database of Information. A recommender system, or a recommendation system (sometimes replacing 'system' with a synonym such as platform or engine), is a subclass of information filtering system that seeks to predict the "rating" or "preference" a user would give to an item.They are primarily used in commercial applications. MOVREC also help users to find the movies of their choices based on the movie experience of other users in efficient and effective manner without wasting much time in useless browsing.

Keywords: Content-based filtering, Recommendation System, Recommender, Collaborative filtering

1. 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 systems 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. Both content-based- and collaborative-based filtering have their own limitations.

Recommendation systems have become well known nowadays, be it in the field of entertainment, education, etc. Earlier, the users needed to settle on choices on what books to purchase, what music to tune in to, what motion pictures to watch and so on. Commercial movie libraries effectively exceed 15 million films, which boundlessly exceeds the visual ability of any single individual. With a large number of motion pictures to browse, individuals now and then get overpowered. Therefore, an efficient recommendation system is necessary for the enthusiasm of both movie service providers and customers. With the improvement of recommendation systems, the customers will have no agony in settling on choices and organizations can keep up their client gathering and draw in new clients by improving user's satisfaction. Additionally, nowadays the modern technologies like machine learning and deep learning also plays a vital role in the process flexible technologies for day to day operations.

2. DEFINITION

There has been a huge increase in audio visual data in the A recommender system, or a recommendation system (sometimes replacing 'system' with a synonym such as platform or

engine), is a subclass of information filtering system that seeks to predict the "rating" or "preference" a user would give to an item. They are primarily used in commercial applications. Recommender systems are utilized in a variety of areas and are most commonly recognized as playlist generators for video and music services like Netflix, YouTube and Spotify, product recommenders for services such as Amazon, or content recommenders for social media platforms such as Facebook and Twitter. These systems can operate using a single input, like music, or multiple inputs within and across platforms like news, books, and search queries. There are also popular recommender systems for specific topics like restaurants and online dating. Recommender systems have also been developed to explore research articles and experts, collaborators, and financial services.

Companies benefit through recommendation system:-

Netflix	2/3rd of the movies watched are recommended
Amazon	35% sales from recommendations
Choice Stream	28% of the people would buy music if they know what they like
Spotify	Recommendations generate 38% more clicl troughs
E-Commerce	With millions of customers and data on their online behavior, e-commerce companies are best suited to generate accurate recommendations
Media	Similar to e-commerce, media businesses are one of the first to jump into recommendations. It is difficult to see a news site without a recommendation system.

3. METHODS

In the field of machine learning, classification methods which use different strategies to organize and classify data. Classifiers could possibly require training data.

A. Collaborative filtering

B. Content-based filtering

C. Multi-criteria recommender systems

D. Risk-aware recommender systems

E. Mobile recommender systems

F. Hybrid recommender systems

A. Collaborative filtering

An example of collaborative filtering based on a ratings system One approach to the design of recommender systems that has wide use is collaborative filtering. Collaborative filtering is based on the assumption that people who agreed in the past will agree in the future, and that they will like similar kinds of items as they liked in the past. The system generates recommendations using only information about rating profiles for different users or items. By locating peer users/items with a rating history similar to the current user or item, they generate recommendations using this neighbourhood. Collaborative filtering methods are classified as memory-based and model-based. A well-known example of memory-based approaches is the user-based algorithm,[24] while that of model-based approaches is the Kernel-Mapping Recommender. A key advantage of the collaborative filtering approach is that it does not rely on machine analysable content and therefore it is capable of accurately recommending complex items such as movies without requiring an "understanding" of the item itself. Many algorithms have been used in measuring user similarity or item similarity in recommender systems. For example, the k-nearest neighbour (k-NN) approach and the Pearson Correlation as first implemented by Allen. When building a model from a user's Behavior, a distinction is often made between explicit and implicit forms of data collection. Examples of explicit data collection include the following: • Asking a user to rate an item on a sliding scale. • Asking a user to search. • Asking a user to rank a collection of items from favourite to least favourite. • Presenting two items to a user and asking him/her to choose the better one of them. • Asking a user to create a list of items that he/she likes (see Rocchio classification or other similar techniques).

Examples of implicit data collection include the following:

- Observing the items that a user views in an online store.
- Analysing item/user viewing times.
- Keeping a record of the items that a user purchases online.
- Obtaining a list of items that a user has listened to or watched on his/her computer.
- Analysing the user's social network and discovering similar likes and dislikes. Collaborative filtering approaches often suffer from three problems: cold start, scalability, and sparsity.
- Cold start: For a new user or item, there isn't enough data to make accurate recommendations.
- Scalability: In many of the environments in which these systems make recommendations, there are millions of users and products. Thus, a large amount of computation power is often necessary to calculate recommendations.
- Sparsity: The number of items sold on major e-commerce sites is extremely large. The most active users will only have rated a small subset of the overall database. Thus, even the most popular items have very few ratings. One of the most famous examples of collaborative filtering is item-to-item collaborative filtering (people who buy x also buy y), an algorithm popularized by Amazon.com's recommender system. Many social networks originally used collaborative filtering to recommend new friends, groups, and other social connections by examining the network of connections between a user and their friends. Collaborative filtering is still used as part of hybrid systems.

B. Content-based filtering

Another common approach when designing recommender systems is content-based filtering. Content-based filtering methods are based on a description of the item and a profile of the user's preferences. These methods are best suited to situations where there is known data on an item (name, location, description, etc.), but not on the user. Content-based recommenders treat recommendation as a user-specific classification problem and learn a classifier for the user's likes and dislikes based on an item's features. In this system, keywords are used to describe the items and a user profile is built to indicate the type of item this user likes. In other words, these

algorithms try to recommend items that are similar to those that a user liked in the past, or is examining in the present. It does not rely on a user sign-in mechanism to generate this often temporary profile. In particular, various candidate items are compared with items previously rated by the user and the best-matching items are recommended. This approach has its roots in information retrieval and information filtering research.

To create a user profile, the system mostly focuses on two types of information:

1. A model of the user's preference.

2. A history of the user's interaction with the recommender system.

Basically, these methods use an item profile (i.e., a set of discrete attributes and features) characterizing the item within the system. To abstract the features of the items in the system, an item presentation algorithm is applied. A widely used algorithm is the tf–idf representation (also called vector space representation). The system creates a content-based profile of users based on a weighted vector of item features. The weights denote the importance of each feature to the user and can be computed from individually rated content vectors using a variety of techniques. Simple approaches use the average values of the rated item vector while other sophisticated methods use machine learning techniques such as Bayesian Classifiers, cluster analysis, decision trees, and artificial neural networks in order to estimate the probability that the user is going to like the item. A key issue with content-based filtering is whether the system is able to learn user preferences from users' actions regarding one content source and use them across other content types. When the system is limited to recommending content of the same type as the user is already using, the value from the recommendation system is significantly less than when other content types from other services can be recommended. For example, recommending news articles based on browsing of news is useful, but would be much more useful when music, videos, products, discussions etc. from different services can be recommended based on news browsing. To overcome this, most content-based recommender systems now use some form of hybrid system. Content-based recommender systems can also include opinion-based recommender systems. In some cases, users are allowed to leave text review or feedback on the items. These user-generated texts are implicit data for the recommender system because they are potentially rich resource of both feature/aspects of the item, and users' evaluation/sentiment to the item. Features extracted from the user-generated reviews are improved meta-data of items, because as they also reflect aspects of the item like meta-data, extracted features are widely concerned by the users. Sentiments extracted from the reviews can be seen as users' rating scores

on the corresponding features. Popular approaches of opinion-based recommender system utilize various techniques including text mining, information retrieval, sentiment analysis (see also Multimodal sentiment analysis) and deep learning.

C. Multi-criteria recommender systems

Multi-criteria recommender systems (MCRS) can be defined as recommender systems that incorporate preference information upon multiple criteria. Instead of developing recommendation techniques based on a single criterion value, the overall preference of user u for the item i, these systems try to predict a rating for unexplored items of u by exploiting preference information on multiple criteria that affect this overall preference value. Several researchers approach MCRS as a multi-criteria decision making (MCDM) problem, and apply MCDM methods and techniques to implement MCRS systems.

D. Risk-aware recommender systems

The majority of existing approaches to recommender systems focus on recommending the most relevant content to users using contextual information yet do not take into account the risk of disturbing the user with unwanted notifications. It is important to consider the risk of upsetting the user by pushing recommendations in certain circumstances, for instance, during a professional meeting, early morning, or late at night. Therefore, the performance of the recommender system depends in part on the degree to which it has incorporated the risk into the recommendation process. One option to manage this issue is DRARS, a system which models the context-aware recommendation as a bandit problem. This system combines a content-based technique and a contextual bandit algorithm.

E. Mobile recommender systems

Further information: Location based recommendation Mobile recommender systems make use of internet-accessing smart phones to offer personalized, context-sensitive recommendations This is a particularly difficult area of research as mobile data is more complex than data that recommender systems often have to deal with. It is heterogeneous, noisy, requires spatial and temporal auto-correlation, and has validation and generality problems. There are three factors that could affect the mobile recommender systems and the accuracy of prediction results: the context, the recommendation method and privacy. Additionally, mobile recommender systems suffer from a transplantation problem – recommendations may not apply in all regions (for instance, it would be unwise to recommend a recipe in an area where all of the ingredients may not be available). One example of a mobile recommender system are the approaches taken by companies such as Uber and Lyft to generate driving routes for taxi drivers in a city. This system

uses GPS data of the routes that taxi drivers take while working, which includes location (latitude and longitude), time stamps, and operational status (with or without passengers). It uses this data to recommend a list of pickup points along a route, with the goal of optimizing occupancy times and profits. Mobile recommendation systems have also been successfully built using the "Web of Data" as a source for structured information. A good example of such system is SMARTMUSEUM The system uses semantic modelling, information retrieval, and machine learning techniques in order to recommend content matching user interests, even when presented with sparse or minimal user data.

F. Hybrid recommender systems

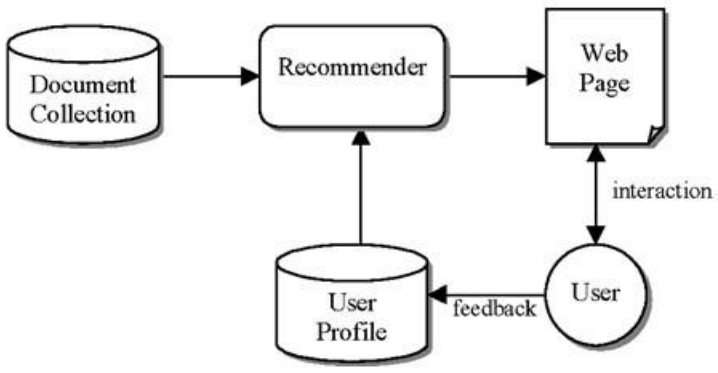
Most recommender systems now use a hybrid approach, combining collaborative filtering, content-based filtering, and other approaches. There is no reason why several different techniques of the same type could not be hybridized. Hybrid approaches can be implemented in several ways: by making content-based and collaborative-based predictions separately and then combining them; by adding content-based capabilities to a collaborative-based approach (and vice versa); or by unifying the approaches into one model (see] for a complete review of recommender systems). Several studies that empirically compare the performance of the hybrid with the pure collaborative and content-based methods and demonstrated that the hybrid methods can provide more accurate recommendations than pure approaches. These methods can also be used to overcome some of the common problems in recommender systems such as cold start and the sparsity problem, as well as the knowledge engineering bottleneck in knowledge-based approaches. Netflix is a good example of the use of hybrid recommender systems. The website makes recommendations by comparing the watching and searching habits of similar users (i.e., collaborative filtering) as well as by offering movies that share characteristics with films that a user has rated highly (content-based filtering)

Some hybridization techniques include:

- Weighted: Combining the score of different recommendation components numerically.
- Switching: Choosing among recommendation components and applying the selected one.
- Mixed: Recommendations from different recommenders are presented together to give the recommendation.
- Feature Combination: Features derived from different knowledge sources are combined together and given to a single recommendation algorithm.

- Feature Augmentation: Co mputing a feature or set of features, which is thee part of the input to the next technique?
- Cascade: Recommenders aree given strict priority, with the lower priority ones breaking ties in the scoring of the higher ones.
- Meta-level: One recommendation technique is applied and produces some sort of model, which is then the input used by the next technique.

4. ARCHITECTURE & CLASSSIFICATION



Many recommendation systems have been developed over the past decades. These systems use different approaches like collaborative approach, content based approach, a utilitty base approach, hybrid approach etc. Looking at the purchase Behavior and history of the shopppers, Lawrence et al. 2001 presented a recommender system which suggests the new product inn the market. To refine the recommendation collaborative and content based filtering approach were used. To find the potential customers most of the recommendation systems today use ratings given by previous users. These ratings are further used to predict and recommend the item of one's choice. In 2007 Weng, Lin and Chen performed an evaluation study which says using multidimensional analysis and additional customer's proffile increases the recommendation quality. Weng used MD recommendation model (multidimensional recommendation model) for this purpose. Multidimensional recommendation model was proposed by Tuzhilin and Adomaavicius (2001).

Technology used in this projectt:-

Framework: Flask
Python
HTML
CSS
JAVA-SCRIPT

5. PROJECT

In this project, we considered thee Movie Lens dataset from the internet, and focus on two files, i.e., the *movies.csv* and *credit.csv*.
Movies.csv has 14 columns whicch define the movie thoroughly:
Like *movie_id*, *budget*, *genre*, etc.

credit.csv has 3 columns:

cast

cre

w id

```
#import necessary libraries
import numpy as np
import pandas as pd
import sklearn
#importing datasets
df1= pd.read_csv('credits.csv')
df2= pd.read_csv('movie.csv')
df1.head()
df2.head()
```

	cast	crew	id
0	[{'cast_id': 14, 'character': 'Woody (voice)',...	[{'credit_id': '52fe4284c3a36847f8024f49', 'de...	862
1	[{'cast_id': 1, 'character': 'Alan Parrish', '...	[{'credit_id': '52fe44bfc3a36847f80a7cd1', 'de...	8844
2	[{'cast_id': 2, 'character': 'Max Goldman', 'c...	[{'credit_id': '52fe466a9251416c75077a89', 'de...	15602
3	[{'cast_id': 1, 'character': "Savannah 'Vannah...	[{'credit_id': '52fe44779251416c91011acb', 'de...	31357
4	[{'cast_id': 1, 'character': 'George Banks', '...	[{'credit_id': '52fe44959251416c75039ed7', 'de...	11862

	budget	genres	homepage	id	keywords	original_language	original_title	overview	popularity	production_comp
0	237000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://www.avatarmovie.com/	19995	[{"id": 1463, "name": "culture clash"}, {"id":...	en	Avatar	In the 22nd century, a paraplegic Marine is di...	150.437577	[{"name": "Ing Film Partners
1	300000000	[{"id": 12, "name": "Adventure"}, {"id": 14, "...	http://disney.go.com/disneypictures/pirates/	285	[{"id": 270, "name": "ocean"}, {"id": 726, "na...	en	Pirates of the Caribbean: At World's End	Captain Barbossa, long believed to be dead, ha...	139.082615	[{"name": "Walt I Pictures", "id":
2	245000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://www.sonypictures.com/movies/spectre/	206647	[{"id": 470, "name": "spy"}, {"id": 818, "name...	en	Spectre	A cryptic message from Bond's past sends him o...	107.376788	[{"name": "Col Pictures", "
3	250000000	[{"id": 28, "name": "Action"}, {"id": 80, "nam...	http://www.thedarkknightrises.com/	49026	[{"id": 849, "name": "dc comics"}, {"id": 853,...	en	The Dark Knight Rises	Following the death of District Attorney Harve...	112.312950	[{"name": "Lege Pictures", "id": 92
4	260000000	[{"id": 28, "name": "Action"}, {"id": 12, "nam...	http://movies.disney.com/john-carter	49529	[{"id": 818, "name": "based on novel"}, {"id":...	en	John Carter	John Carter is a war-weary, former military ca...	43.926995	[{"name": "Walt I Pictures", "

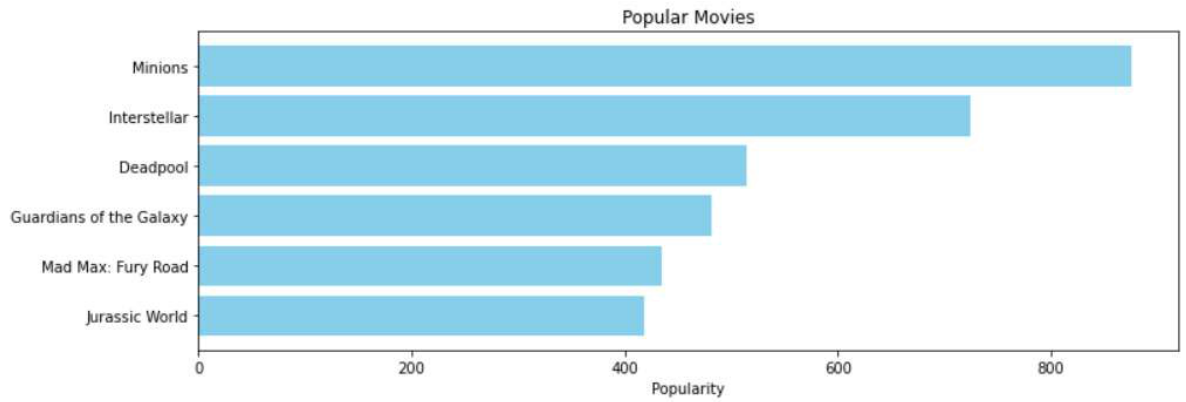
Output

#define a new feature 'score' and calculate its value with weighted_rating()
#Print the top 15 movies

q_movies[['title', 'vote_count', 'vote_average', 'score']].head(10)

	title	vote_count	vote_average	score
1881	The Shawshank Redemption	8205	8.5	8.059258
662	Fight Club	9413	8.3	7.939256
65	The Dark Knight	12002	8.2	7.920020
3232	Pulp Fiction	8428	8.3	7.904645
96	Inception	13752	8.1	7.863239
3337	The Godfather	5893	8.4	7.851236
95	Interstellar	10867	8.1	7.809479
809	Forrest Gump	7927	8.2	7.803188
329	The Lord of the Rings: The Return of the King	8064	8.1	7.727243
1990	The Empire Strikes Back	5879	8.2	7.697884

 # We have made our first(though very basic) recommender.
#Under the Trending Now tab of these systems we find movies that are very popular
#and they can just be obtained byy sorting the dataset by the popularity column.



CONTENT BASED FILTERINNG

Using TfidfVectorizer ,we Create a TF-IDF matrix here . The 'stop_words' paramm tells the TF-IDF module to ignore common english words like 'the', etc.

```
from sklearn.feature_extraction.text import TfidfVectorizer

tfidf= TfidfVectorizer(stop_words='english')

# Replace Nan with an empty string
df2['overview']=df2['overview'].fillna('')

# construct the required TF-IDF matrix by fitting and transforming the data
tfidf_matrix= tfidf.fit_transform(df2['overview'])

tfidf_matrix.shape
```

Then we compute similarity betwween all products using SciKit Leanr's linear_kernel (which in this case is equivalent to cosine similarity).

```
from sklearn.metrics.pairwise import linear_kernel

#compute the cosine similarity matrix
cosine_sim= linear_kernel(tfidf_matrix,tfidf_matrix)
```

- Now in this system the content of the movie (overview, cast, crew, keyword, tagline etc) is used to find its similarity with other movies.
- Then the movies that are most likely to be similar are recommended.

```
df2['overview'].head(5)
```

```
0    In the 22nd century, a paraplegic Marine is di...
1    Captain Barbossa, long believed to be dead, ha...
2    A cryptic message from Bond’s past sends him o...
3    Following the death of District Attorney Harve...
4    John Carter is a war-weary, former military ca...
Name: overview, dtype: object
```

We are going to define a function that takes in a movie title as an input and outputs a list of the 10 most similar movies. Firrstly, for this, we need a reverse mapping of movie titles and Data Frame indices.

Example:-

Input - *get_recommendation('The Dark Knight Rises')*

Output-

```
65                              The Dark Knight
299                              Batman Forever
428                              Batman Returns
1359                                    Batman
3854    Batman: The Dark Knight Returns, Part 2
119                              Batman Begins
2507                                 Slow Burn
9           Batman v Superman: Dawn of Justice
1181                                      JFK
210                              Batman & Robin
Name: title, dtype: object
```

Input-

```
get_recommendation('The Avengers')
```

output

```
7               Avengers: Age of Ultron
3144                             Plastic
1715                             Timecop
4124                This Thing of Ours
3311             Thank You for Smoking
3033                     The Corruptor
588     Wall Street: Money Never Sleeps
2136         Team America: World Police
1468                         The Fountain
1286                          Snowpiercer
Name: title, dtype: object
```

RESULT:-

CONCLUSION

Recommender Systems have emerged as powerful tools for helping users find annd evaluate items of interest. These systems use a variety of techniques to help users identify the items that best fit their tastes or needs. While poopular CF-based algorithms continue to prodduce meaningful, personalized results in a variety of domains, data mining techniques are increassingly being used in both hybrid systems, to improove recommendations in previously successful applications, and in stand-alone recommenders, too produce accurate recommendations in previoously challenging domains. The use of data miningg algorithms has also changed the types of recoommendations as applications move from recommmending what to consume to also recommending when to consume. While recommender systems may have started as largely a passing novelty, they clearly appear to have moved into a real and powerful tool in a variety of applications, and that data mining algorithms can be and will continue to be an important part of the recommendation process.