

Microsoft Project



Algorithms

(Movie Recommendation)

Submitted By:

Simranjeet Kaur

Institute: Dr. B.R. Ambedkar National Institute Of Technology, Jalandhar

Course: B.Tech

Branch- CSE

Mentor: Soumya Tiwari

Movie Recommendation System

Introduction

Recommender systems aim to predict users' interests and recommend product items that quite likely are interesting for them.

As per the business standpoint, the more relevant products a user finds on the platform, the higher their engagement. it results in increased revenue for the various platform.

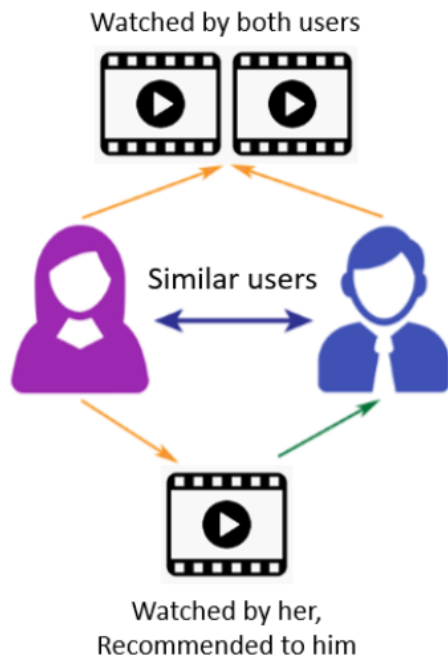
Types or techniques of Recommendation Systems:

- Content-Based Recommendation Systems
- Collaborative Filtering Recommendation Systems

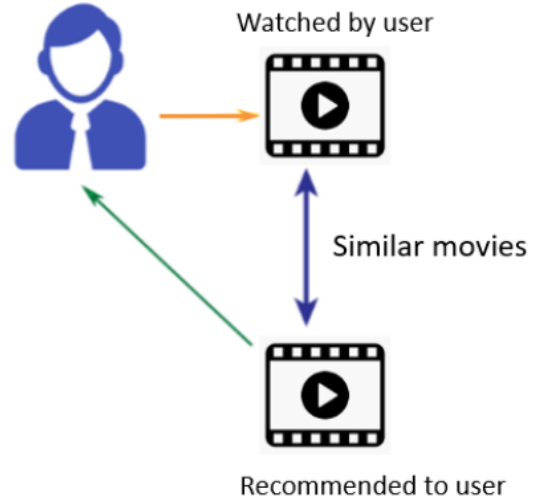
Modern recommendation uses the both approaches called as Hybrid recommendation.

For my project, I have used content-based recommendation Algorithm.

Collaborative Filtering



Content-Based Filtering



Dataset Information

Dataset of the project contains 20 attributes:

budget,genres,homepage,id,keywords,original_language,original_title,overview,popularity,production_companies,runtime,spoken_languages,status,tagline,title,vote_average,vote_count,movie_id,cast,crew

But from these attributes we will use only these attributes:

'movie_id','title','overview','genres','keywords','cast','crew'

After the data pre-processing and cleaning we will have only three attributes that are: movie_id, title, tags(consisting of overview, genres, keywords, cast, crew)

Problem Statement

Our aim is to recommend the user 5 best movies according to the one-movie chosen by the user. There is total of 5000 approx. movies in our dataset.

Algorithms Used: Content-Based Recommendation

Data Cleaning and Pre-processing

The first step of my data pre-processing process is to handle the missing data. Then we have just collected the useful keywords from the attributes and combined them in single attribute called 'Tags'.

Modeling and Training

For modeling purpose, bag of words technique is used. Best 5000 keywords have been taken based on the frequency of words and then each movie is vectorized based on that 5000 keywords. To recommend 5 best movies related to a movie, cosine distance is measured. If distance is less, similarity will be high and if distance is more, similarity will be less.

```
In [25]: from sklearn.metrics.pairwise import cosine_similarity
```

```
similarity = cosine_similarity(vector)
```

```
similarity
```

```
Out[25]: array([[1.          , 0.08964215, 0.06071767, ..., 0.02519763, 0.0277885 ,
 0.          ],
 [0.08964215, 1.          , 0.06350006, ..., 0.02635231, 0.          ],
 [0.06071767, 0.06350006, 1.          , ..., 0.02677398, 0.          ],
 ...,
 [0.02519763, 0.02635231, 0.02677398, ..., 1.          , 0.07352146,
 0.04774099],
 [0.0277885 , 0.          , 0.          , ..., 0.07352146, 1.          ,
 0.05264981],
 [0.          , 0.          , 0.          , ..., 0.04774099, 0.05264981,
 1.          ]])
```

Results:

```
recommend('Gandhi')
```

Gandhi, My Father
The Wind That Shakes the Barley
A Passage to India
Guiana 1838
Ramanujan

References

- https://www.youtube.com/watch?v=1xtrIEwY_zY
- <https://medium.com/analytics-vidhya/movie-recommendation-system-aa204f0700eb>
- <https://www.toptal.com/algorithms/predicting-likes-inside-a-simple-recommendation-engine>

THANK YOU!!