Movie Recommender Report

Peter Waysocher, Max Jakobitsch, Alexander Widmann

June 2020

1 Introduction

The goal is to create a system that recommends movies based on different strategies. We use Django and Bootstrap to build the website, where the user can choose a movie and gets recommendations based on it and the algorithm chosen. The methods work with Pandas to efficiently edit the data and Sklearn for the machine learning algorithm. TMDb API provides us with more metadata for the movies. The project is designed to be easily adaptable. So we can add or adjust strategies without problems.

2 Methods

2.1 User-Based

We implemented two user-based strategies. Both share the same principles. They collect all movies from the users which rated the base movie and then calculate the average rating for each movie.

2.1.1 Filter out below-average ratings

This method filters out all ratings which are below the average. So it uses only positive propagation to generate the recommendations.

2.1.2 Popularity-based

Here all ratings are used. Furthermore, the final score is calculated based on the average ratings and their popularity.

2.2 Contend-Based and Hybrids

2.2.1 Genre-based

The genre-based method iterates through all movies and calculates the number of genres that both possess. Afterward the system recommends the five highest ones. Furthermore, this method has the option to includes the ratings and popularity of the movies. Therefore, the average rating, popularity and the similar genres are used to calculate the score.

2.2.2 Keywords only

2.2.3 Meta-mix

This method works similar to the genre-based method. Additionally to the matching genres, the numbers of matching actors, directors, keywords and production countries are calculated. These numbers are divided through the highest occurring number and multiplied with a multiplier based on importance. Furthermore, the year difference is calculated and inversed.

Component	Multiplier
Genres	35%
Actors	20%
Directors	8%
Keywords	10%
Production countries	7%

After the calculations each movie has a score between 0 and 100 and the five highest ones will be recommended. This method has an option for popularity too and works similar to the genre-based one.

2.2.4 First direct sequel + co

2.2.5 Mixed Algorithms

2.2.6 Mixed Algorithms with popularity bias

2.3 Evaluation

For the evaluation, we used 22 different movies such as Star Wars, Toy Story or Pirates of the Caribbean. These movies have all different genres and popularity. We score the recommendations between 1 and 10 based on the look and feel, title, cover, summary, genres and knowledge. One is the lowest score and ten the highest. We had to choose so many factors because no one of us is an expert on this topic. So the scoring is amateurish and should be taken with a grain of salt. Furthermore, we gave prequels and sequels less score if there are many. Everyone scored them alone and then we calculated the average for each method.

2.3.1 User-based

We tried to evaluate and score both user-based methods like the others but after some iterations, we gave up on scoring them. The problem is the recommended movies from the "filtered-below-average" are more or less random. They don't really share similarities with the base movie and are solely chosen based on their good ratings. Furthermore, the results were mostly unpopular movies that nobody knows of. On the other hand, the popularity based method recommends often the same movies which are popular and have a high rating.

2.3.2 Content-based and Hybrids

Rank	Method	Score
1	Meta-mix	6.74
2	Genre-based	6.17
3	Mixed Algorithms	6.16
4	First direct sequel + co	6.14
5	Mixed Algorithm with popularity bias	6.06
6	Keywords only	5.41

Meta-mix is the winner in our evaluation, but the second place is interesting too. Similar genres play a more important role for good recommendations than keywords. This is probably because we are not experts and decides the score more based on matching genres than the rest. But only looking at the genres don't give the best results. It is best to choose a tactic that combines many factors, but genres play a huge role. "Keywords only" scores the lowest. This is due to our knowledge of this subject. Keywords will probably play a higher role if you know the movies.

2.3.3 Additionally observations

Anne Frank is a biography of her life. It has the genres "Drama, Foreign, Documentary". Interesting is that the methods using genres present more documentary than the methods which are based on keywords.

A problem was the movies which have a big universe for example "Marvel universe" or "Lord of the Rings". The recommendations often were movies from the universe which are no prequel or sequel and matched pretty well with the base movie. These movies scores are higher than other movies. On the other hand, unpopular movies have a low average score. The reason is that methods that uses popularity struggle to provide good recommendations and that the results are often unpopular movies which we could not score probably. A funny observation is that the methods based on keywords sometimes recommend movies which do not match well. For example, the method "Mixed Algorithm" recommends the horror movie "Deathship" for "Titanic". This result is probably archived by the keywords "ship" or "shipwreck".

Animations or Children movies have the highest score. It was easy to evaluate and it got a higher score because of it.