

# Advanced Movie Recommender

## Gelişmiş Film Öneri Sistemi

Selim Sırrı Sevgi, Ömer Özgür Bozkurt  
Computer Engineering Department  
Yıldız Technical University  
İstanbul, Turkey  
11110007@std.yildiz.edu.tr  
ozgur@ce.yildiz.edu.tr

**Abstract**—Nowadays with the help of advancement in technology, the productivity of the film industry is increasing faster than ever. Also people have more movie options in their hands thanks to different methods for digital consuming and delivering in recent years. All these changes may sound positive, but they bring different problems with them as well. Such is that the more movies a person has to pick from, the more trouble it may cause. For example, spending more time on picking a movie when they could enjoy watching one. In this project, we try to focus on this problem and use computer technology to help people to find and decide to watch movies without any regret that they would feel in case of wasting their time on a movie, which is not their taste.

**Keywords** — *movie recommending system; recommending system; recommending engine, movie recomender.*

**Özetçe**—Teknolojideki gelişmelerin yardımıyla günümüzde film endüstrisi piyasaya daha fazla ve hızlı film sürüyor. Ayrıca son yıllarda medya tüketim ve ulaştırma araçları arttıkça insanların izleyebilecekleri filmlerin sayısı da artıyor. Bu gelişmeler olumlu gözükmesine rağmen bazı problemlere de sebep olmaktadır. İzlenebilecek filmlerin sayısı arttıkça, izlenecek bir filmin seçilmesi de zorlaşacaktır. Bir filmi izlemek yerine o zamanı izlenecek bir film bulmak için harcıyor olabilirler. Bu projede bu problemin üzerine yoğunlaşarak ve bilgisayar teknolojilerini kullanarak insanların zevkleri olmayan bir filmi izledikten sonra zamanlarının boşa gittiği düşüncesinin getireceği pişmanlık olmadan filmler bulmasına yardımcı olmaya çalışıyoruz.

**Anahtar Kelimeler** — *film öneri sistemi; öneri sistemi; öneri motoru, film önericisi.*

- **Introduction:** For every kind of item, a problem arises when there are lots of them to pick from. These items could be a book, a movie, a song etc. The improvements of different methods provide ways to produce them faster and result in huge amount of them. One could realize that there must a

way to let people survive in this no-ending stream of items. A lot of big companies and scientist deal with this problem in different fields and in different ways. Such companies would be Netflix [1] for movies, Spotify [2] for music and Amazon [3] for various products. Netflix had put lots of energy and spent lots of money in order to improve its recommending system [4]. Netflix stands as the best example of what we tried to achieve. However, these system do their job well, they have some drawbacks. They are big and complicated system for such a basic task. There is a need for a simple and practical tool in order to give the chance to people discover what they would like to watch and offer them suggestions to make progress easier and faster. This project's main goal is to help people finding movies for their taste easily without wasting too much time. Even though our system is more beneficial, it does not require people to register for the service. It does not try to do many things, its main focus to give people a user-friendly interface to discover movies, and help them find movies for their taste and preferences.

- **System Design:** The fundamental data in the system is movie. Therefore, a way to populate database with movies had been needed. It has been decided to use data given on IMDB(Internet Movie DataBase) website [5]. Afterwards IMDbPY [6] tool had been used to transform text based data into a database. It had been observed that the database did not have enough data such as ratings, runtime etc. However, this database had been useful for having a list of movies. So the list had been used to do queries though OMDb(Open Movie DataBase) [7] in order to have all possible information about each movie. It had been decided to work on small set of movies, which was more than a million entries. The system consists three main moduls which power the recommendation engine. One of is calculation of the similarities of the movies. This gives power to our system for having

recommendations for guest users. Another one is the advanced search functionality. It provides a way for users to discover movies based on comprehensive filters. The last and most important one is make recommendations based on user's activities. These activities are ratings, commenting, searching and creating different lists. The most important part of the system design had been developing the recommendation engine. It processes all data collected from users and does complicated calculation based on some powerful developed algorithms in order to create recommendations for each user.

It has been decided to implement the Project on Ruby on Rails [8]. The reasons for this preference has been the simplicity of the framework, strong support in the community, the willingness to keep up with the latest technology. The system was baked up with MySQL, it is able to handle big load and scale if necessary. It is the proven choice for a website project.

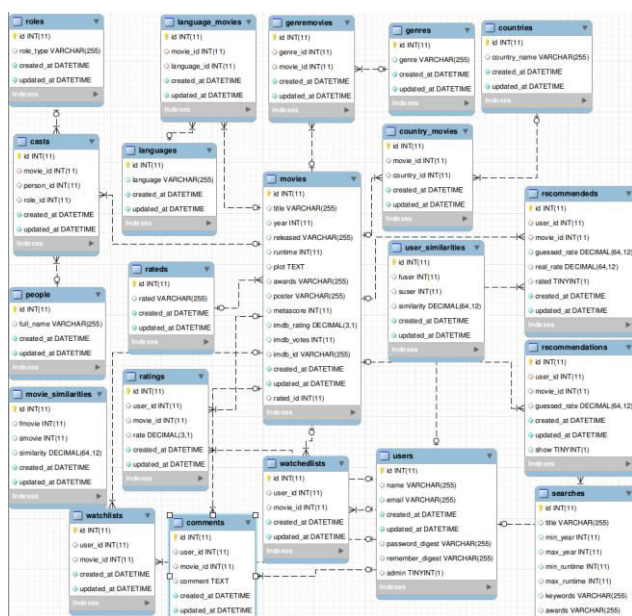


Figure 1. ER Diagram

Figure 1 shows the structure of the database. It had been designed to avoid repetition of data, to use storage in the best way possible.

- **Experimental Study:** One of the important aspect in the project had been made the system as easeir as possible to use. A guest user can simple take advantage of advanced searching to find movies for their mood and criteria for any moment. As they go through movies, the system recommends them other similar movies. With this functionality the user can see all the details of a movie and discover movies they would not know or would not be able to find by just searching. The comment area of a movie is going to help them to judge the movie based on other people's opinion. Even though it is not a requirement to register, definitely it is going be more beneficial for users to keep the history of their activities by creating lists. Registered users are also capable of commenting on movies and rating them. These kinds of actions are going to be used by the system to personalized the recommendations more.

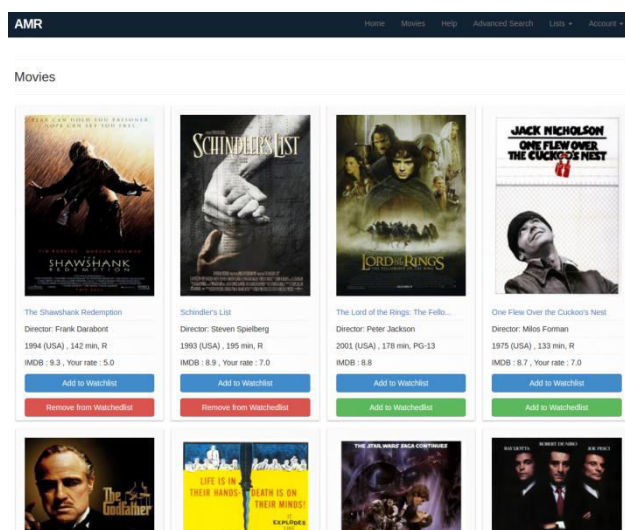


Figure 2. Movies Screen

Figure 2 shows general look of movie listing interface. Each movie is represented in a movie card. This principles was followed on every possible page in order to have consistency.

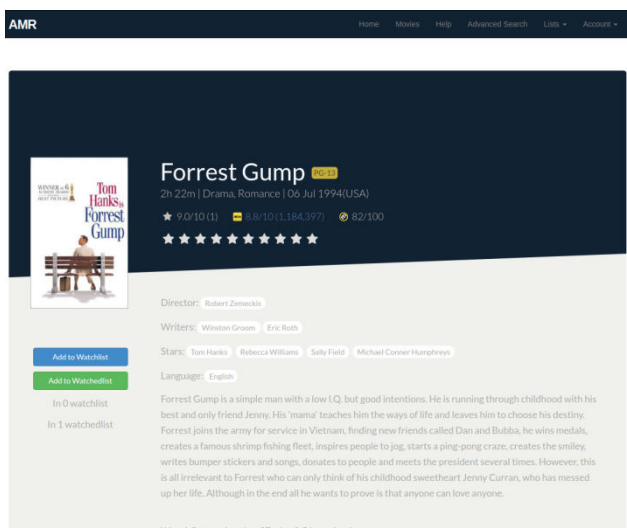


Figure 3. Movie Details Screen

Figure 3 is the page that show a movie's details. All available information about a movie is represented on this page. Logged in users can rate the movie, add it either to their watchlist or watchedlist.

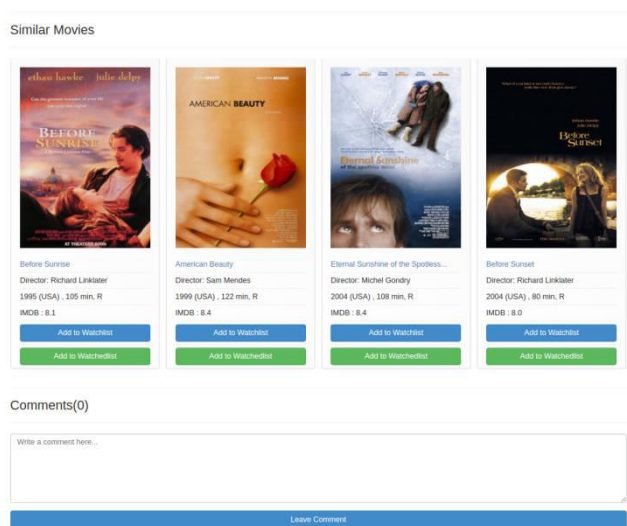


Figure 4. Movie Details Screen-2

Figure 4 shows the bottom of the same movie detail page. Similar movies to the one that is being viewed is located right under the movie details. This is a powerful feature that gives the opportunity to users to discover new movies. More down to the page, the comment area can be seen. Here all users can read the available reviews about the movie and get a better idea on what the movie is all about.

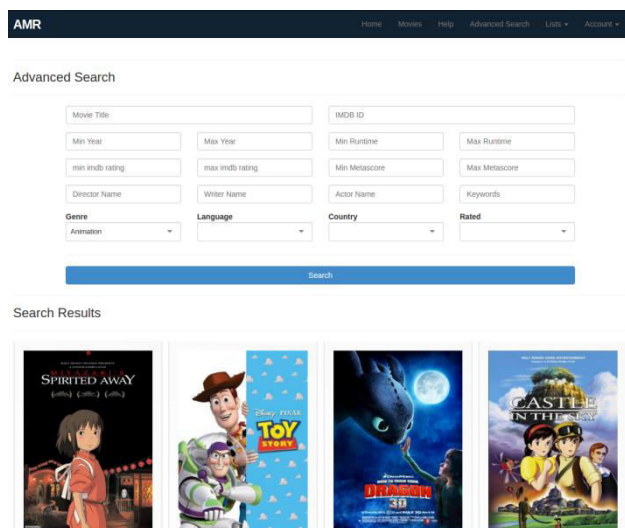


Figure 5. Advanced Search Screen

Figure 5 shows the advanced search page. With the help of the filters on this page, users can find movies easily based on their criteries.

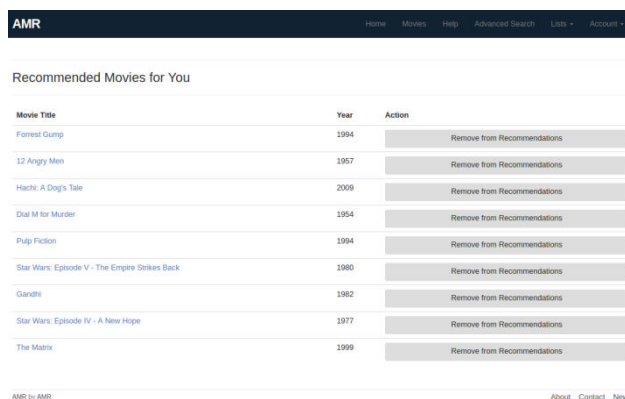
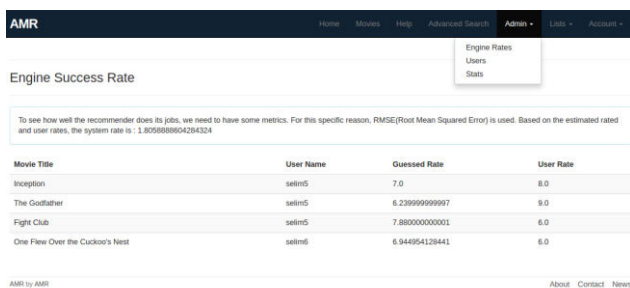


Figure 6. Recommendations Screen

Figure 6 is the page showing the system recommendations for the user. The user may check out the movies and perform such as rating, commenting, adding to the lists. Afterwards, the movies may be removed from the list for further processing in order to measure the system.



**Figure 7.** Monitoring Screen

Figure 7 shows the menu for an authorized access and metrics for monitoring the recommendation engine. An authorized user can also view the user list and delete them if necessary, see the statistics on movie count, user count etc.

- Results and Discussion:** The outcome of the project had been satisfying. The recommendation engine uses all available data from the user to offer them some movies that they would like. The system is not just a constantly listing movies for the user but also taking into consideration of what the user has done with the each movie that was recommended. The system is learning more and more by each input from the user. This input could be count as negative or positive. The system keeps tracks of every action for further processing. This also helps to measure its performance. The manager could review the success of the system from the managing panel. Our study has brought a simple but effective movie recommendation system. Unlike other tools, its strength is in its simplicity. The

power of the engine comes from combining all techniques and taking advantages of each.

- Conclusion:** The more movies and the more ways of watching movies are available, the more it becomes problematic. Therefore, the study had been about this problem and creating a way to overcome it. The approach had been forming a set of movies first, then providing ways for users to discover new movies for their preferences manually and also by the developed recommendation engine. The results and feedback from the test group had been mostly positive. There is no an uplimit for the success rate of the engine, so the algorithms can always be improved more to cover more parameters. There could be more ways to offer a guest user recommendations. However, it is not an easy task because of a cold start.

- References:**

- [1] "Netflix", <https://www.netflix.com/tr/>. 15-Apr-2016.
- [2] "Spotify", <https://www.spotify.com/tr/>. 20-Apr-2016.
- [3] "Amazon", <http://www.amazon.com/>. 23- Apr -2016.
- [4] "Netflix Prize", <http://www.netflixprize.com/>. 23-May-2016.
- [5] "Alternative Interfaces", <http://www.imdb.com/interfaces/>. 23-May-2016.
- [7] "IMDbPY", <http://imdbpy.sourceforge.net/>. 26-May-2016.
- [8] "Ruby on Rails", <http://rubyonrails.org/>. 26-May-2016.
- [9] "MySQL", <https://www.mysql.com/>. 26-May-2016.