CSC478 Final Project Report
Jun Tae Son | Yuchen Wu | Tzu Hao Peng

Movielens Recommender System Analysis

1. Introduction

With nowadays advanced technology, we can stream movies easily from websites like Netflix, Hulu, Amazon, etc. There are more than thousands of movies provide for users to enjoy daily. Instead of browsing around and wasting time to look for what movies we should watch next, a movie recommendation system will suggest movies that you might like. A great movie recommender can enhance our entertainment and lifestyle. The system can suggest movies based on users interest, movie contents, genres and popularities. However, we all have different tastes and judgements of each movie. A personalized movie recommender system is the best application to suggest movies based on user's preference. It is also the new driven force for companies to generate enormous revenues. There are many methodologies behind this sophisticated engine, such as content based, collaborative filtering, and hybrid filtering approach. Our analysis is to explore these methods and discover the highlights. We also studied several research papers to get a better understanding of this powerful engine and how to assure the quality of a recommendation system.

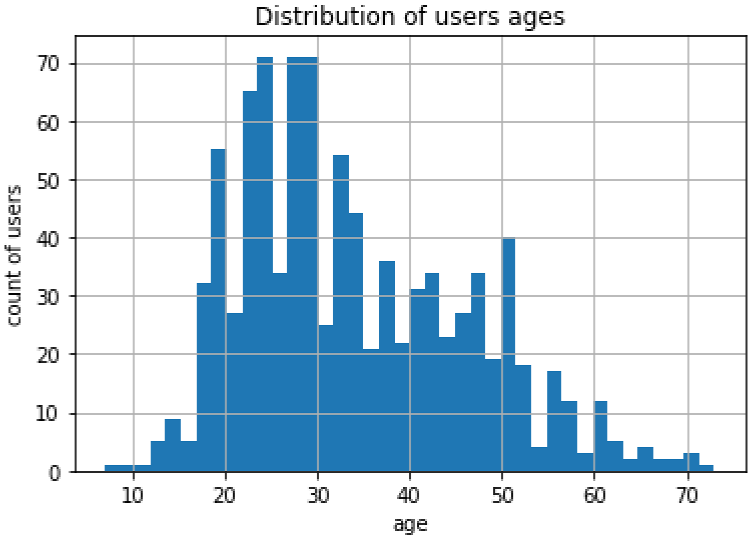
2. Data description and Analysis

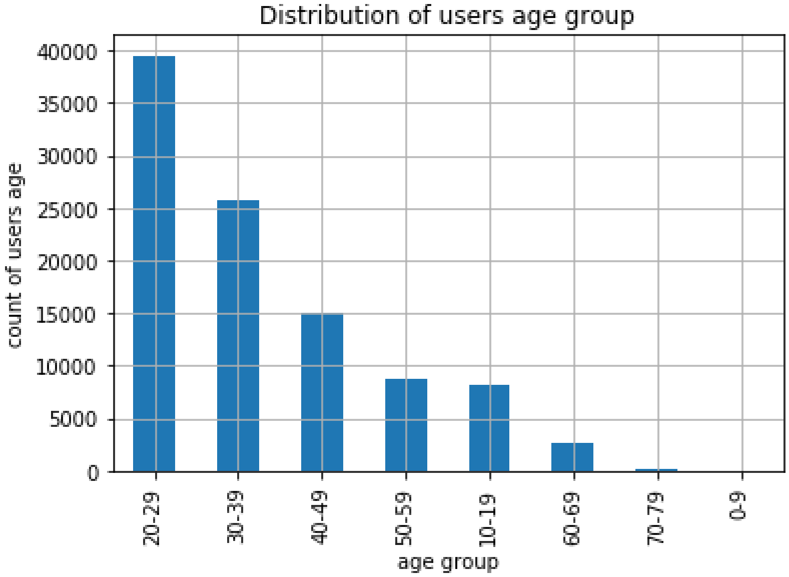
2-1 Data Source: we used movie ratings data from www.movielens.org. The data set contains 100,000 ratings (scale 1-5) from 943 users on 1682 movies. Each user has rated minimum 20 movies.

- u.data file contains user id, movie id, rating and timestamp
- u.item file contains movie id, movie title, release date, video release date, imdb url, unknown, action, adventure,animation,childrens, comedy, crime, documentary, drama, fantasy,film-noir,horror,musical,mystery, romance,sci-fi,thriller,war,western.
- u.user file contains user id, age, gender,occupation,zip code
- u.genre contains a list of the genres

2-2 General analysis:

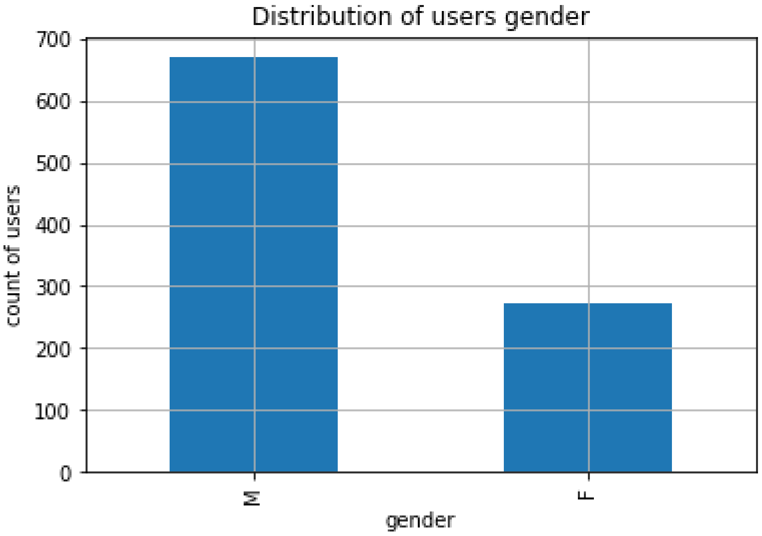
We imported four files for the general analysis. The average age of people contributing to the rating is 34 and the average movie score is 3. There are 10 movies have average rating score 5. However, neither of these are popular movies. In fact, there are few users rated these movies in result of the perfect score. The most rated movie is Star Wars and there are 583 users who rated this movie and the average rating score is 4.4. Besides, we also explored the data to see the rating characteristics in different movie genres. Film-Noir movies have the highest rated score 3.9. Unknown and fantasy have the lowest rated scores compared to other genres. As you can see below, we used histogram and bar graphs to visualize the distribution of ages.

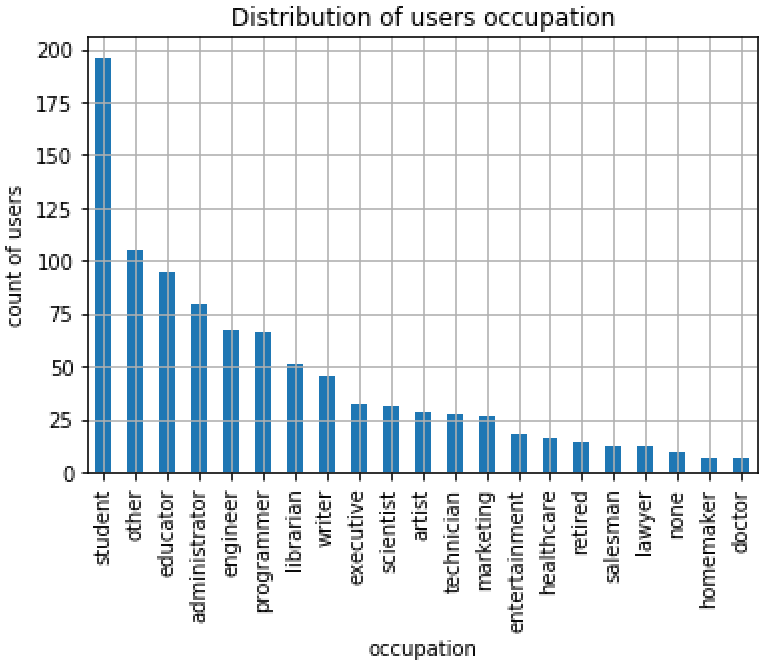




The age group from 20 to 29 years are the most people participated in the rating survey and they have the lowest average rating score 3.5. The millennials are the ones criticized more about the movies than other age groups. One possible reason is that they are savvy about mass media and love to give feedbacks. Age from 0 to 9 gives the highest average rating score 3.8, but they only represented a small population.

Furthermore, there are more male participants than females in this dataset. The average rating score by males and females are almost the same. The average rating table includes 100 movies and rated by male and females. *2001:A Space Odyssey* is highly rated by males compare females and *Sound of Music* is favored by females the most.





Last but not least, the occupation of the most users in this dataset are students, and doctors are the least people who contribute to the survey. The average rating score for people who don't have an occupation give the highest average rating score (3.7), and people who work in healthcare field give the lowest average rating score (2.6).

3. Techniques used in Proposed Methodology

3-1. Content-based filtering

Content-based recommender system gives recommendations that are similar in content of the items. This approach uses the item's description and user's preference to explore the previously rated items and recommends items of the nearest neighbors. Manoj Kumar et al.

explains that one of the drawbacks of content-based filtering is to require users to answer several questions related to their preference information, thus it may take a lot of time if one uses many features/attributes that describe an item to build an algorithm. However, the results based on the content-based filtering tend to be highly relevant to a user's preference, and new items can be recommended quickly.

In our project, we built a basic content-based movie recommendation algorithm to recommend movies with highest similarity according to the movie genres. We computed a dot product of movie genre with ratings to generate a user profile matrix. The user profile explains user's preference towards each movie genre. Then we calculated Pearson similarity to a user's profile and generated a list of similar movies that are closest in similarity.

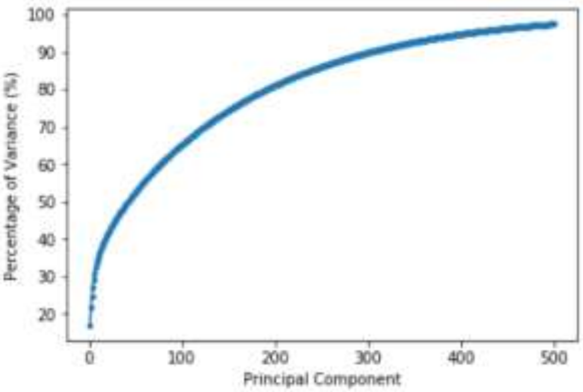
3-2. Collaborative filtering

Unlike content-based filtering that measures *similarity in content* such as user's preference and item's description, collaborative filtering recommender system is based on *similarity measure between users and items*. Manoj Kumar et al. gives a few advantages of collaborative filtering: It depends on connection between users and items only, for example user ratings across items. These explicit ratings can measure a real quality of items. But content-based filtering is not useful when a new item is introduced and there is no rating on it.

User-based collaborative filtering

This approach recommends an item based on user group similarity. For example, if user A's preference is similar to the group of users B, then the algorithm returns a list of items that users in group B highly rated.

We used k-mean clustering for user-based collaborative filtering recommendation. We created the matrix by using the number of users and number of items to represent an approach of recommending movies. PCA is a way to reduce dimensions and we use 300 components instead of the original dataset 943 and the components represent 90% of the total variance. Besides, we used the result to do k-means clustering and set k = 300. Based on users' rating, each clusters represent 3 movies with 5 genre types. However, user-based k-means is not a good approach because the data is too sparse and many cluster size are only 1 which means only one user in the cluster.



Item-based collaborative approach

This approach recommends a new item that is similar to the items that a user has shown interest. D. Zhao et al. proposed an assumption in item-based collaborative filtering that a user will like an item A if the user highly rated an item B, and both item A and B share similar properties. In this paper, we proposed a item-based movie recommendation algorithm based on KNN and SVD. Firstly, we created pivot matrix of movie-ratings to be one row per user and one column per movie, and normalized it by subtracting the mean. For KNN approach, we computed distance of movie ratings based on Pearson similarity measure and retrieved 5 most closest movies. For SVD, we predicted movie ratings from the decomposed matrices and compared the distances between the actual and the predicted.

4. Evaluation result

We used RMSE and MAE to evaluate our models, because they represent forecasting errors of the models between the predicted and actual values. We compared 10 different collaborative filtering algorithms through 5-folds cross-validation test as shown in the table below. The result shows that KNN based and SVD based approach give better results among others.

<5-folds cross-validation test>

Algorithm	RMSE	MAE
NormalPredictor	1.5181	1.2180
BaselineOnly	0.9441	0.7484
KNNBasic	0.9785	0.7725
KNNWithMeans	0.9505	0.7489
KNNBaseline	0.9300	0.7326
SVD	0.9367	0.7379
SVD++	0.9189	0.7205
NMF	0.9626	0.7560
SlopeOne	0.9447	0.7427
CoClustering	0.9663	0.7567

We proposed three different movie recommender algorithms as we discussed above; content-based filtering, item-based filtering with KNN and SVD. We tested those algorithm with a sample who has user_id 1, and retrieved a list of five movies and their genres for each model. For content-based filtering algorithm, we got five different movies with the same genre. So, we can assume that user 1 will enjoy watching comedy, drama, and romance genre movies. For item-based collaborative filtering with KNN and SVD approaches, the recommended movies from each model were different one another, but the movie genres were similar.

< Comparison of 5 recommended movies from each algorithm for user_id=1 >

Algorithm	**Recommended movies**
Content-based filtering	['Comedy', 'Drama', 'Romance'] Cinema Paradiso (1988) ['Comedy', 'Drama', 'Romance'] Wings of Desire (1987) ['Comedy', 'Drama', 'Romance'] Manhattan (1979) ['Comedy', 'Drama', 'Romance'] American President, The (1995) ['Comedy', 'Drama', 'Romance'] Corrina, Corrina (1994)
Item-based collaborative filtering with KNN	['Action', 'Comedy', 'Western'] Young Guns (1988) ['Drama'] Three Colors: White (1994) ['Comedy', 'Western'] City Slickers II: The Legend of Curly's Gold (1994) ['Drama'] Glengarry Glen Ross (1992) ['Comedy', 'Romance', 'Thriller'] Ghost (1990)
SVD collaborative filtering	['Action', 'Crime', 'Thriller'] Heat (1995) ['Drama'] An Unforgettable Summer (1994) ['Comedy'] Friday (1995) ['Thriller'] Assassins (1995)

	['Romance'] Kissed (1996)

5. Conclusion

Overall, the average movie rating score is 3. There are 10 movies are rated at 5; however, there are only few users rated those movies. Star Wars is the most popular rated movie, it has 583 users and rated at 4.4. There are more male participants than females in the dataset. Age from 20-29 are the most people rated in the survey;however, it has the lowest average rating score. One potential reason is the millennials are savvy about mass media and they are more harsh on the rating score. Students are the most people who participate in the survey and doctors are the least category. People who don't have an occupation tend to give the highest average rating score and people who are in the healthcare field give the lowest average rating score.It can be explained that people who don't have an occupation will have more free time to enjoy watching movies and give ratings.

Furthermore, k-means approach is not a good way to build recommender system due to the sparse data and the large feature selections. Based on the PCA feature selections, some of the cluster size are only 1, which means some clusters only include 1 user's preference.

In this paper, we compared three different movie recommender algorithms and the recommendation results. We selected one feature 'genre' to generate a simple recommender algorithm, and we were able to extract the favorite genre of movie from each user. To build an item-based filtering approach, we retrieved the most similar items based on K nearest neighbors. Eventually, we found that all three different recommender algorithms recommended movies with similar genre. But the lists of 5 recommended movie results from each model were different one another. Since each model is based on different approaches, it is reasonable that they do not provide the same result. We expect that implementation with weighting method suggested by Manoj Kumar et al. and larger dataset with less sparsity may help the recommender systems achieve a better performance.

Reference:

Manoj Kumar, D K Yadav, Ankur Singh and Vijay Kr. Gupta. Article: A Movie Recommender System: MOVREC. International Journal of Computer Applications 124(3):7-11, August 2015. Published by Foundation of Computer Science (FCS), NY, USA.

D. Zhao, J. Xiu, Y. Bai and Z. Yang, "An improved item-based movie recommendation algorithm," 2016 4th International Conference on Cloud Computing and Intelligence Systems (CCIS), Beijing, 2016, pp. 278-281.

S. Agrawal and P. Jain, "An improved approach for movie recommendation system," *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, Palladam, 2017, pp. 336-342.

Bokde, Dheeraj kumar et al. “Role of Matrix Factorization Model in Collaborative Filtering Algorithm: A Survey.” CoRR abs/1503.07475 (2015): n. pag.