MovieLens Community Network:
Movie Recommender System

Presented by: Harshit Arora, Frank Fan, Yaping Zhang, Anuraaj Sonawala

Directed by: Prof. Demetrius Lewis

December 8, 2019

Introduction and Research Question

Ever since Netflix started providing 'on-demand' video services in 2007, the online video streaming industry has not only brought in $125 billion in revenue but also taken over other models such as Cable and Cinemas. With so much content on the internet, one of the biggest problems consumers face nowadays is what to actually watch. With so many choices available, consumers are almost burdened with choices, and providers must aid consumers by recommending the 'right' content to the right viewer. MovieLens, a web-based recommender system is one platform that helps solve this issue. Run by GroupLens, a research lab at the University of Minnesota, the virtual community recommends movies for users to watch based on their film preferences using collaborative filtering of these members' movie reviews and ratings. Collaborative filtering is a technique that looks for patterns in user's ratings in order to produce recommendations. It works by searching through a large group of people and finding a smaller subset of users with similar tastes. By combining the items both parties like, it creates a list of movie suggestions. The success of MovieLens can be measured by the fact that their users have contributed 11M movie ratings and regularly utilize this platform to decide which movie they should watch next, and because their dataset has constantly been rated as one of the best[1] for recommendation systems research.

When learning about their service, our first thought was How does MovieLens make recommendations and what factors are being considered when making these recommendations? More specifically, our research focuses on understanding how we can leverage movie-movie networks and/or user-user networks to measure similarity, and whether this similarity measure can make accurate recommendations to users. We will go on to utilize user-based collaborative filtering (UBCF) during the process, instead of content-based filtering, which relies on movie

[1] https://www.scss.tcd.ie/joeran.beel/blog/2019/08/03/and-the-winner-is-movielens-on-the-popularity-of-recommender-system-datasets/

characteristics such as genre/release year. UBCF is a widely used recommendation system approach and is the chosen recommendation technique of MovieLens. In practice, a hybrid recommendation system approach is also widely used by streaming services such as Netflix2.

Background and Data Description

The dataset we are using for this project is from the MovieLens website which contains 963 users and 1682 movies. Variables about users such as UserID, Age, Gender, Occupation etc as well as films (MovieID, Movie Title, Genre etc) are available. Moreover, the dataset contains 100,000 Ratings, rated on a scale from 1-5. 17,000 of these ratings have a low rating of either a 1 or a 2, 27,000 have a neutral value of 3 and the majority (55,000) have a rating of 4 or 5.

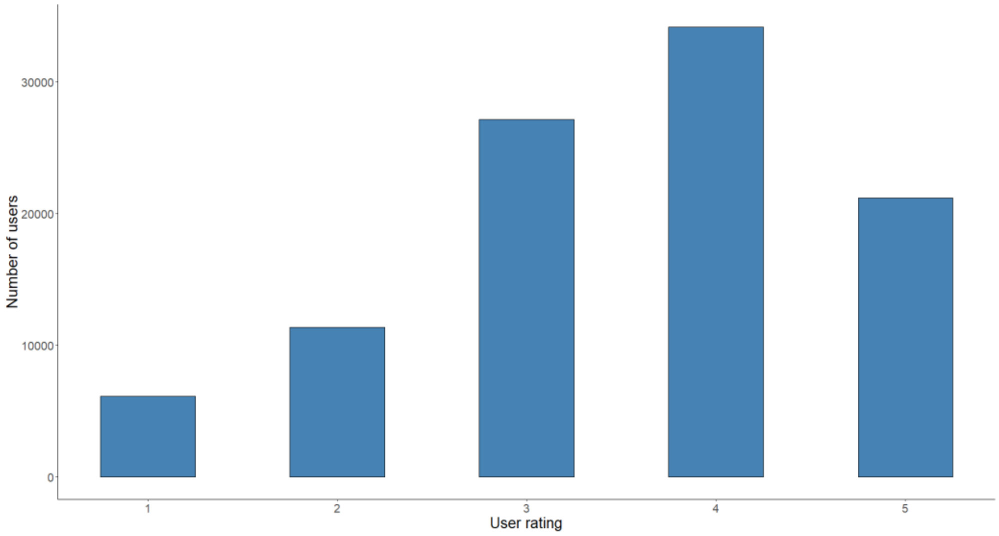
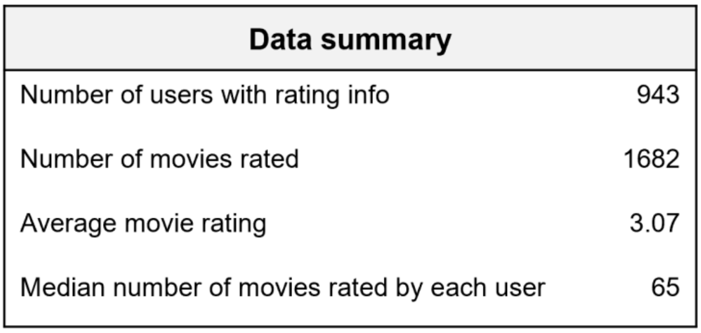


Figure-1 User ratings distribution

The user distribution of ratings shows a variation in rating tendencies of users. Majority of the movies have received a rating of 3 and above, and each movie has received a median 65 average 107 number of ratings. To analyze our data, we will transform the user movie ratings data to

[2] https://help.netflix.com/en/node/100639

form adjacency and incidence matrixes, and consequently analyze the networks and calculate similarity measures, which will form the backbone of UBCF.



Data summary	
Number of users with rating info	943
Number of movies rated	1682
Average movie rating	3.07
Median number of movies rated by each user	65

Table-1 Data Overview and Summary

The two networks in our scope are the Movie based (Movie-Movie) network and User based (User-User) network. In a movie-based network, movies are connected if they are rated by the same user. Here, we are looking at if there are any specific genres that are likely to be rated higher by its users or if having a higher centrality helps movies get a higher rating. Conversely, in a User based network, Users are connected if they have similar sentiment about the same movie. Sentiment here is defined by having either a relatively low rating (1,2) or a high rating (4,5). Some of the questions we are looking at here are who the most central users are, do users who like the same movies tend to dislike the same movies, identifying which users are similar to each other based on similarity measures such as the Cosine similarity, which can be used to make recommendations of the Top-N rated movies from a set of most similar users for each individual user.

Movie-based network

In the movie-based network, we recognized ties between movies if they have been rated by the same user. The weight of edges in this network indicates how many users have watched and rated the same movie. Why do we care about the movie-based network? In order to build a movie recommender system, we need to understand what characteristics of movies contribute to high ratings. The first possible answer is the genre of movies. Second, the position of a movie in the movie network might also influence its rating. The following analyses unfold in these two aspects to explore the relationship between ratings and certain movie characteristics.

1. Are movies in certain genres more likely to be rated higher by users?

We have run a linear regression predicting the ratings of movies as a function of movie genres. It is likely that the number of ratings the movie has received will affect its final rating: if a movie only has few rating, then a biased unfair low rating might cause a huge influence on the overall rating; however, if the number of raters is huge, then the effect of extreme unfair ratings can be smoothed out. Therefore, in the regression the number of ratings the movie has received was added as a control variable. We also include each movie's release year and the timestamp of ratings as control variables in the regression.

Based on the regression result[3], even after we control the number of ratings each movie has received, certain genres of movies still receive better ratings than others. For example, action, children and horror movies are more likely to receive lower ratings. By contrary, documentary and drama movies are more likely to receive higher ratings.

One possible explanation behind this result is that people generally have higher expectations for action, children and horror movies and are more likely to be strict or even picky towards movies in these genres. Another possible explanation is that in current film industry, documentary and drama movies have more mature production system, more advanced production technology and higher production standard, therefore can produce better quality movies which naturally get higher ratings. Since movies genre carries significant weights in movie rating, we need to carefully take it into consideration while doing further movie rating analysis or building the recommender system.

[3] Appendix 1: Regression predicting ratings as movie genres

2. Whether being at the center of the movie network leads to higher ratings?

In order to answer this question, different centrality measures were used to determine which movies are in the center of the network. The advantage of choosing five centrality measures is that different centrality measures capture different information in terms of network center and analyzing them all can provide better insights and further help to determine which one is the best measure of network center.

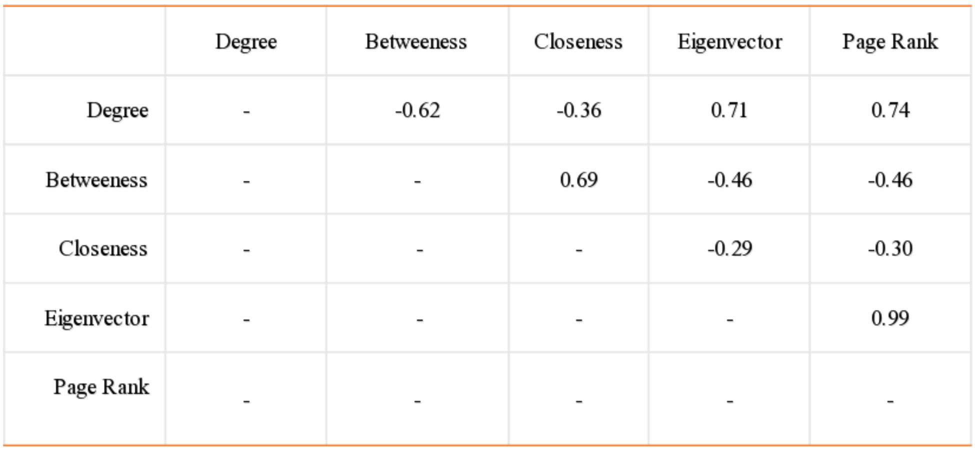
The correlations between each centrality measure and movie ratings indicates how strongly each centrality measure relate to movie rating and the direction of their relationship. Besides correlation, a better and more thorough approach is to conduct regression analysis. Therefore, we have run five regression predicting movie ratings using the five centrality measures respectively. And just as in part one, release year, rating time and number of ratings the movie has received are added in the regression model as control variables.

	Closeness	Betweenness	Degree	Eigenvector Centrality	Page Rank
Rating	**-0.09**	**-0.16**	**0.22**	**0.25**	**0.25**

Table-2.1 Correlation of movie ratings with various centrality measures

Based on the results of correlation analysis and regression analysis[4], we found that closeness and betweenness have negative relationships with movie rating while degree, eigenvector centrality and page rank have positive relationships with movie rating. One explanation behind this seemingly contradictory result is that in the movie-based network, different centrality measures define network center in different ways and therefore capture different network centers. To verify this idea, we have computed the correlations between each pair of centrality measures, and the result is shown on the next page.

[4] Appendix 2: Regression between five centrality measures and ratings in the movie-based network



	Degree	Betweeness	Closeness	Eigenvector	Page Rank
Degree	-	-0.62	-0.36	0.71	0.74
Betweeness	-	-	0.69	-0.46	-0.46
Closeness	-	-	-	-0.29	-0.30
Eigenvector	-	-	-	-	0.99
Page Rank	-	-	-	-	-

Table-2.2 Correlation of five centrality measures in the movie-based network

From the result, closeness and betweenness are positively related while degree, eigenvector centrality and page rank are positively related to each other, but the relationship between the two groups of centrality measures is negative. Therefore, we can define two types of centers in the movie-based network: one captured by closeness and betweenness, the other captured by degree, eigenvector centrality and page rank.

The two different types of centers in movie network have different characteristics. First, movies with high degree can be defined as ‘genre films’. These movies have been co-watched with many similar movies by the same audience who have similar tastes and thus have high degree (Marvel serial movies, Star Wars, godfather). These movies tend to attract their specific audience and receive high ratings. As a result, network center defined by degree is positively related to ratings.

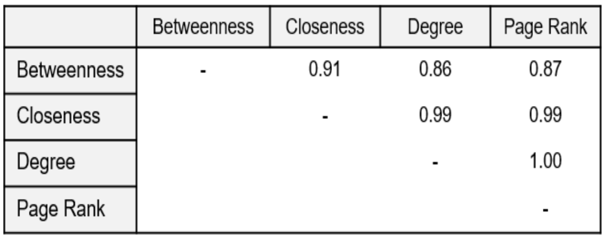
In the contrary, movies with high betweenness are the movies bridging different groups of users with different tastes. In another word, these movies have been co-watched by many users regardless of their usual preferences. They get lower ratings probably because they are more likely to be popcorn movies that try to appeal to everyone’s tastes and needs, therefore more likely to fail (Squeeze, Boys in Venice).

3. Conclusion

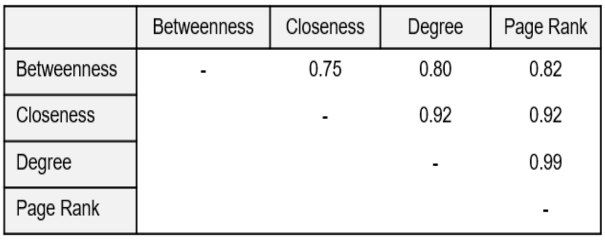
Overall, we have gained enough insights to build a movie-based recommender system. However, movie recommender systems built on movie-based network is only able to find similarities among movies and give users similar movie recommendations to the ones that already highly rated by users. This is to say, it will only recommend users movies that are similar to those he already rated high and it is not capable to find new movies which a user has not seen in this type or genre before. Therefore, to overcome this limitation and provide a functional movie recommender system, we decided to dive deeper and continue exploring the user-based network.

User-User Network

The user-user network will be the backbone of a collaborative filtering-based recommendation system. Users are connected if they have similar sentiments about the same movie. This criterion is crucial, as it ensures that an edge is formed between 2 users only if they have rated the same movie high (4 or 5) in the like network, or low (1 or 2) in the dislike network.



	Betweenness	Closeness	Degree	Page Rank
Betweenness	-	0.91	0.86	0.87
Closeness		-	0.99	0.99
Degree			-	1.00
Page Rank				-



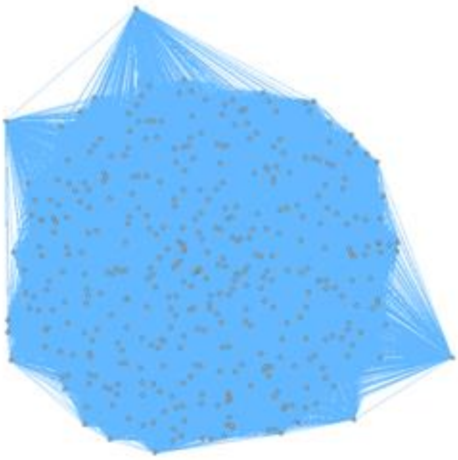
	Betweenness	Closeness	Degree	Page Rank
Betweenness	-	0.75	0.80	0.82
Closeness		-	0.92	0.92
Degree			-	0.99
Page Rank				-

Table 3.1: (left) Centrality measure correlations of User-User like network

Table 3.2: (right) Centrality measure correlations of User-User dislike network

A centrality measure analysis on the two networks showed that our primary centrality measures are highly correlated with each other, indicating that similar sets of users are expected to be more central regardless of the measure of centrality chosen. Intuitively, users with high closeness, for example, have more movies that were rated similarly to other users in the overall network, which shows that their tastes are similar to a lot of other users.

Analyzing these two networks separately also raised a question: Do users who like the same movies also tend to dislike the same movies? Our analysis revealed that there is a 92.9% overlap between the edges of the like and dislike networks, a crucial insight that strengthens the belief of a successful similarity-based recommendation system.



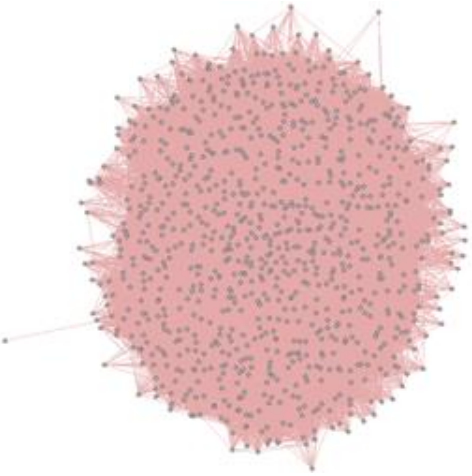


Figure 2.1: (left) User-User like network of 400 users randomly sampled out of 963

Figure 2.2: (right) User-User dislike network of the sampled 400 users

1. User demographics’ impact on ratings[5]

We also wanted to understand if people with certain demographics (age, gender, or occupation) tend to rate higher than others. In order to control the effect of varying number of movies for each genre and varying tastes in different years of release, we used genre and year as control variables while predicting the movie ratings. Results showed that males tend to give lower ratings than females. Certain occupations, such as Writers and Healthcare professionals, have tendencies to rate lower than others as well.

[5] Appendix 3: Regression between movie ratings and demographics in the user-user network

2. Measuring similarity between users[6]

We have established that certain user characteristics have an impact on their tendency to rate a movie higher or lower. What we also want to do is to measure the similarity between any pair of users. We rely on Jaccard similarity as a measure to summarize the similarity of a user to other users in the network based on the like and dislike network of users. Multidimensional scaling is performed to attain four co-ordinates (two from each network) and the four co-ordinates are used as control variables in the demographic regression analysis. This showed a similar result, i.e, males tend to rate lower than females, while professionals in the entertainment industry rate higher than others. With this process and insights in mind, a need and potential success of a similarity-based recommendation system is reinforced.

Building Movie Recommender Systems

1. Identify potential core–periphery structure

To build a movie recommender system, the first step is to identify whether there is a core-periphery structure in the user-based networks. Essentially, core-periphery structure has a densely connected set of network nodes in the center, and many sparsely connected, non-central network nodes in the periphery. In the MovieLens user-based networks, the existence of a core-periphery structure will indicate that a core of users shares similar sentiment towards the same movies, and in the periphery, there are some users who don’t share similar movie tastes with others.

Why do we care about the core-periphery structure in the user-based networks? If there are clear cores in the user-based networks, it will be difficult to establish one single recommender system for all the users because users in the periphery tend to have distinct

[6] Appendix 4: Regression between movie ratings and demographics in the user-user network using multi-dimensional scaling similarities as controls

tastes, thus the recommender system could generate unsuitable recommended movies for them. Presence of a core-peripheral structure might indicate the need for separate recommender systems for users who have "mainstream" preference of movies and users who have a "niche" preference of movies.

We utilized three descriptive evidence to demonstrate the structure of the user-based networks:

1) Closeness:

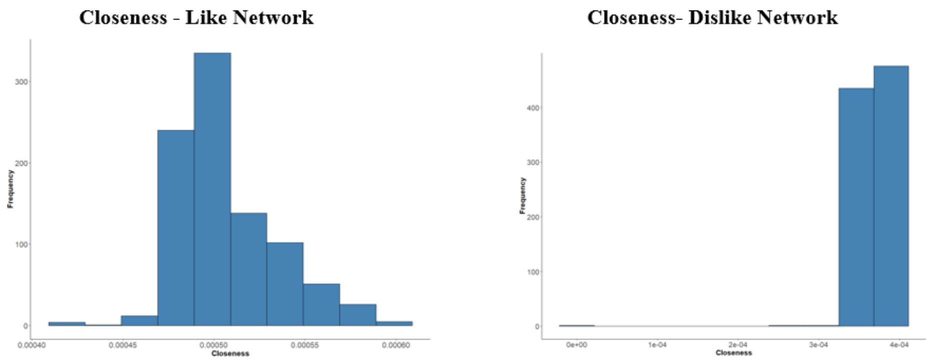
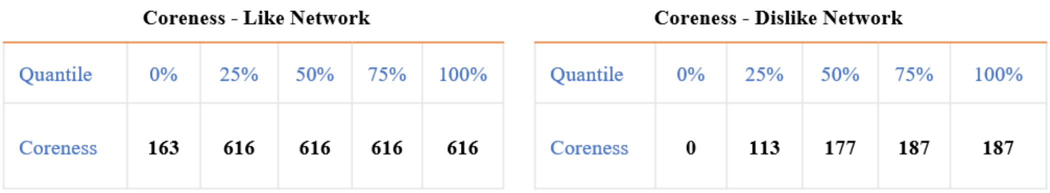


Figure 3.1: Closeness distribution in the user-based networks

If there is a core-periphery structure, closeness will be sparsely distributed because users in the periphery tend to significantly low closeness. In both the like-network and the dislike-network, lack of dispersion in the distribution of closeness indicates that a clear core-periphery structure does not exist.

2) Coreness distribution:



Coreness - Like Network

Quantile	0%	25%	50%	75%	100%
Coreness	**163**	**616**	**616**	**616**	**616**

Coreness - Dislike Network

Quantile	0%	25%	50%	75%	100%
Coreness	**0**	**113**	**177**	**187**	**187**

Table 3.2: Coreness distribution in the user-based networks

After calculating how many nodes are included in different levels of k-cores, we found that more than 25% of nodes in the like network are in k-core with degree 616 and more than 25% of nodes in the dislike network are in k-core with degree 177. As a result, the majority of users have high and similar coreness, indicating the absence of a clear core-periphery structure.

3) Largest connected component:

   We decomposed both the like network and the dislike network into their largest connected component and checked whether the largest components contain most of the users in the networks. The largest connected component of the like network contains 100% of users, and the largest connected component of the dislike network contains 99.8% of users, which again solidifies the lack of nodes in the periphery.

Conclusion: all three pieces of descriptive evidence exhibit that the user-based network does not have a clear core-periphery structure and most of the users are well connected. Thus, a single recommender system would suffice in the MovieLens community network.

2. Cosine similarity movie recommender[7]

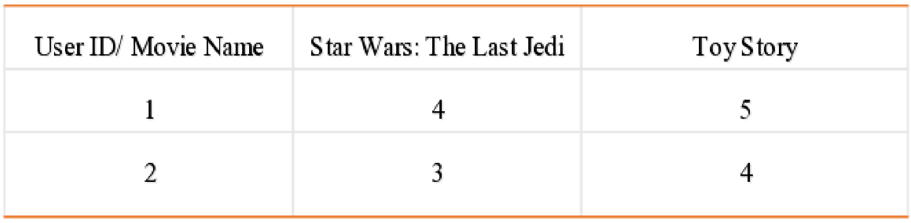
The cosine similarity between two non-zero vectors is a similarity measure that calculates the cosine of the angle between them. The cosine similarity between two users in the user-based network can represent similarity of movie preferences.

We have built a movie recommender based on cosine similarity with the following steps:

1) Calculate the cosine similarity between each pair of users in the network

$$cosine\ similarity\ = \frac{\alpha \cdot \beta}{||\alpha|| \cdot ||\beta||}$$

[7] Appendix 5: Sample code for user-based collaborative filtering



User ID/ Movie Name	Star Wars: The Last Jedi	Toy Story
1	4	5
2	3	4

Table 3.3: Example of cosine similarity between two users

For example, both user 1 and user 2 have rated Star Wars: The Last Jedi and Toy Story. We can represent each user's ratings as vectors: $\alpha = [4,3], \beta = [5,4]$, and get the cosine similarity$= \frac{(4\times5)+(3\times4)}{\sqrt{4^2+3^2}\times\sqrt{5^2+4^2}} = 0.99$. In this example, the two users have very similar movie preferences because the cosine similarity is close to the maximum value of 1.

2) Find the five users that have the highest cosine similarity for each user
3) Recommend the five top ranked yet not rated movies, based on average ratings from similar users identified in step two, to each user

By splitting rating data into training and test set, we have tested the cosine similarity movie recommender. And the recommended movies will receive an average rating of 4.21 from users, which proved the viability of such a similarity-based movie recommender.

3. User-based collaborative filtering with 'Recommenderlab'[8]

An alternative approach is to use the UBCF algorithm from the R 'Recommenderlab' package. The UBCF algorithm takes rating data by multiple users for multiple items as input and directly generates a top-N recommendation list for a given user. It is more convenient to use the 'Recommenderlab' package' because it can adjust the number of recommendations easily according to provider/user needs.

[8] Appendix 6: Sample code for user-based collaborative filtering with "recommenderlab"

$$Accuracy = \frac{Correct\ recommendations}{Total\ recommendations}$$

For the evaluation of the movie recommender, the data was divided into training and validation and accuracy rate was used to show how many of the movies recommended in top-N lists were rated high by users (with rating 4 or 5). As a result, the recommender built with UBCF function to create a top 5-recommendation list for each user has an accuracy rate of 84.3%.

Summary

1. Approaches and results

Using user movie ratings data from MovieLens, we have constructed two types of networks: movie-based network and user-based network. With the combination of network analysis, correlation analysis and regression techniques, we have three major findings:

1) Certain genres of movies tend to receive higher ratings, whereas certain demographic characteristics of users are related to higher ratings too.
2) The dense structure of the user-based network makes the usage of one single movie recommender system pragmatic and feasible.
3) Similarity measures can reflect the similarity of movie preferences and thus helps us establish a movie recommender system.

2. Implications

Based on the understanding of network characteristics and structure, we have used two different methods to establish movie recommenders for MovieLens users. The first method deploys cosine similarity to find the most similar users in terms of movie tastes and create recommendations. The second method directly utilizes the UBCF function from the "Recommenderlab" package and generates a Top-N recommendations list for users. Both recommenders perform well in validation. With our findings and recommenders, MovieLens can further improve their movie recommendation mechanism and create value for users.

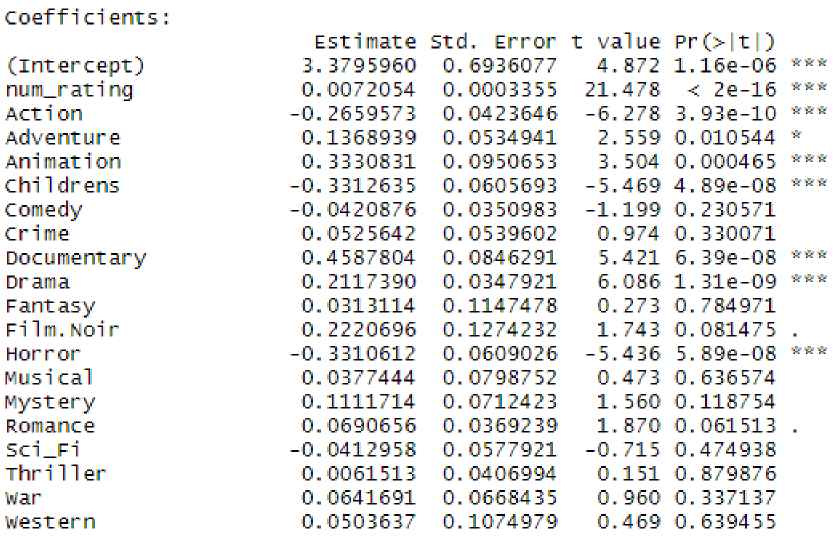
3. Limitations and further recommendations

One major limitation is the data size we have used is relatively insufficient compared to the size of movie industry. Ideally, we would test this recommendation system on a much wider set of users' ratings on a much larger movie universe. With more data in hand, we would like to observe the performance as well as computational efficiency of different recommender systems (such as Item-Item collaborative filtering, which is known for generating rather stable models) and select one that is both computationally cheap and provides highly accurate recommendations that users choose to view and rate high.

Due to data constraints, we were not able to recognize the distinction between a user choosing to watch a movie and choosing to rate a movie either. For instance, users only rate a subset of the movies that they have watched. Taking the movies watched but not rated into account can help us account for a clear distinction and make tighter recommendations, which should inherently increase the performance of UBCF. That said, since choosing to rate a movie tends to indicate more stronger feelings, we are still able to make recommendations a high accuracy.

Appendix

1. Regression predicting ratings as movie genres



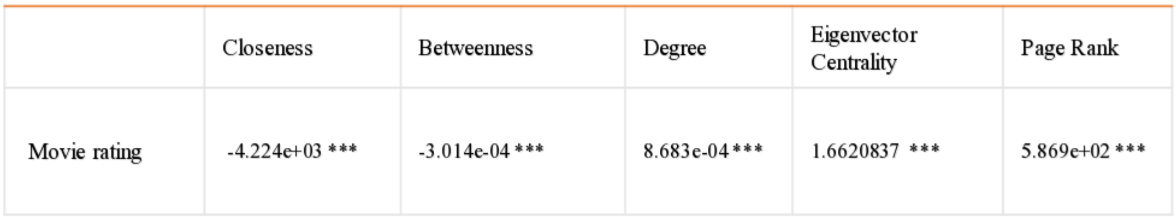
```
Coefficients:
                  Estimate Std. Error t value Pr(>|t|)    
(Intercept)      3.3795960  0.6936077   4.872 1.16e-06 ***
num_rating       0.0072054  0.0003355  21.478  < 2e-16 ***
Action          -0.2659573  0.0423646  -6.278 3.93e-10 ***
Adventure        0.1368939  0.0534941   2.559 0.010544 *  
Animation        0.3330831  0.0950653   3.504 0.000465 ***
Childrens       -0.3312635  0.0605693  -5.469 4.89e-08 ***
Comedy          -0.0420876  0.0350983  -1.199 0.230571    
Crime            0.0525642  0.0539602   0.974 0.330071    
Documentary      0.4587804  0.0846291   5.421 6.39e-08 ***
Drama            0.2117390  0.0347921   6.086 1.31e-09 ***
Fantasy          0.0313114  0.1147478   0.273 0.784971    
Film.Noir        0.2220696  0.1274232   1.743 0.081475 .  
Horror          -0.3310612  0.0609026  -5.436 5.89e-08 ***
Musical          0.0377444  0.0798752   0.473 0.636574    
Mystery          0.1111714  0.0712423   1.560 0.118754    
Romance          0.0690656  0.0369239   1.870 0.061513 .  
Sci_Fi          -0.0412958  0.0577921  -0.715 0.474938    
Thriller         0.0061513  0.0406994   0.151 0.879876    
War              0.0641691  0.0668435   0.960 0.337137    
Western          0.0503637  0.1074979   0.469 0.639455    
```

Certain genres of movies eceive better ratings than others. For example, action, children and horror movies are more likely to receive lower ratings. By contrary, documentary and drama movies are more likely to receive higher ratings.

2. Regression between five centrality measures and ratings in the movie-based network

Regression model: rating ~ centrality measure + number of ratings + genre + factor (release year) + factor (rate year)

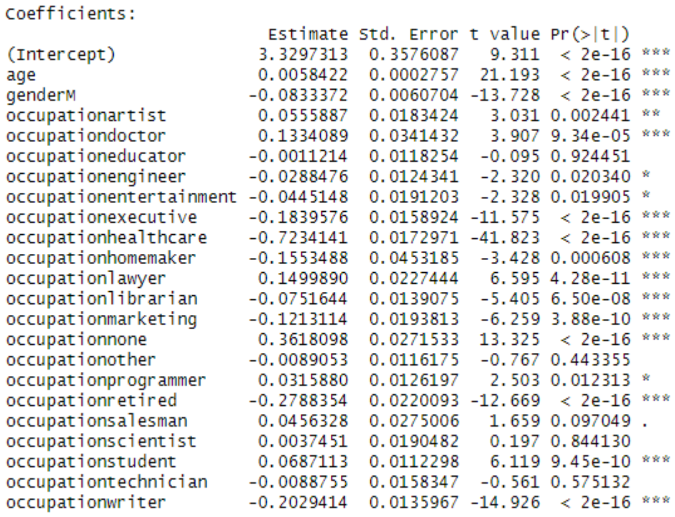
Regression coefficients:



	Closeness	Betweenness	Degree	Eigenvector Centrality	Page Rank
Movie rating	-4.224e+03 ***	-3.014e-04 ***	8.683e-04***	1.6620837 ***	5.869e+02 ***

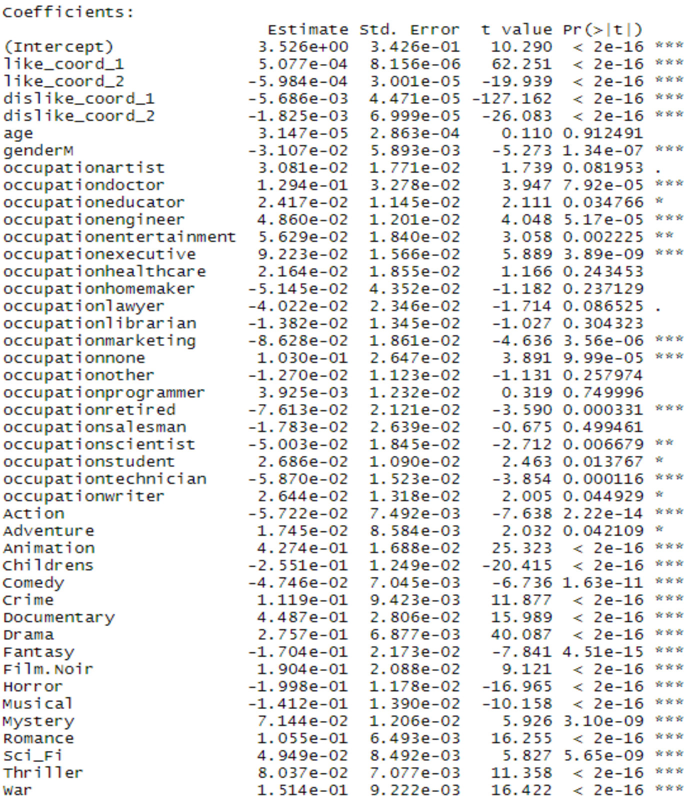
Closeness and betweenness have negative coefficients with movie ratings while degree, eigenvector centrality and page rank have positive coefficients with movie ratings.

3. Regression between movie ratings and demographics in the user-user network



```
Coefficients:
                          Estimate Std. Error t value Pr(>|t|)    
(Intercept)              3.3297313  0.3576087   9.311  < 2e-16 ***
age                      0.0058422  0.0002757  21.193  < 2e-16 ***
genderM                 -0.0833372  0.0060704 -13.728  < 2e-16 ***
occupationartist         0.0555887  0.0183424   3.031 0.002441 ** 
occupationdoctor         0.1334089  0.0341432   3.907 9.34e-05 ***
occupationeducator      -0.0011214  0.0118254  -0.095 0.924451    
occupationengineer      -0.0288476  0.0124341  -2.320 0.020340 *  
occupationentertainment -0.0445148  0.0191203  -2.328 0.019905 *  
occupationexecutive     -0.1839576  0.0158924 -11.575  < 2e-16 ***
occupationhealthcare    -0.7234141  0.0172971 -41.823  < 2e-16 ***
occupationhomemaker     -0.1553488  0.0453185  -3.428 0.000608 ***
occupationlawyer         0.1499890  0.0227444   6.595 4.28e-11 ***
occupationlibrarian     -0.0751644  0.0139075  -5.405 6.50e-08 ***
occupationmarketing     -0.1213114  0.0193813  -6.259 3.88e-10 ***
occupationnone           0.3618098  0.0271533  13.325  < 2e-16 ***
occupationother         -0.0089053  0.0116175  -0.767 0.443355    
occupationprogrammer     0.0315880  0.0126197   2.503 0.012313 *  
occupationretired       -0.2788354  0.0220093 -12.669  < 2e-16 ***
occupationsalesman       0.0456328  0.0275006   1.659 0.097049 .  
occupationscientist      0.0037451  0.0190482   0.197 0.844130    
occupationstudent        0.0687113  0.0112298   6.119 9.45e-10 ***
occupationtechnician    -0.0088755  0.0158347  -0.561 0.575132    
occupationwriter        -0.2029414  0.0135967 -14.926  < 2e-16 ***
```

4. Regression between movie ratings and demographics in the user-user network using multi-dimensional scaling similar as controls



```
Coefficients:
                          Estimate Std. Error  t value Pr(>|t|)    
(Intercept)              3.526e+00  3.426e-01   10.290  < 2e-16 ***
like_coord_1             5.077e-04  8.156e-06   62.251  < 2e-16 ***
like_coord_2            -5.984e-04  3.001e-05  -19.939  < 2e-16 ***
dislike_coord_1         -5.686e-03  4.471e-05 -127.162  < 2e-16 ***
dislike_coord_2         -1.825e-03  6.999e-05  -26.083  < 2e-16 ***
age                      3.147e-05  2.863e-04    0.110 0.912491    
genderM                 -3.107e-02  5.893e-03   -5.273 1.34e-07 ***
occupationartist         3.081e-02  1.771e-02    1.739 0.081953 .  
occupationdoctor         1.294e-01  3.278e-02    3.947 7.92e-05 ***
occupationeducator       2.417e-02  1.145e-02    2.111 0.034766 *  
occupationengineer       4.860e-02  1.201e-02    4.048 5.17e-05 ***
occupationentertainment  5.629e-02  1.840e-02    3.058 0.002225 ** 
occupationexecutive      9.223e-02  1.566e-02    5.889 3.89e-09 ***
occupationhealthcare     2.164e-02  1.855e-02    1.166 0.243453    
occupationhomemaker     -5.145e-02  4.352e-02   -1.182 0.237129    
occupationlawyer        -4.022e-02  2.346e-02   -1.714 0.086525 .  
occupationlibrarian     -1.382e-02  1.345e-02   -1.027 0.304323    
occupationmarketing     -8.628e-02  1.861e-02   -4.636 3.56e-06 ***
occupationnone           1.030e-01  2.647e-02    3.891 9.99e-05 ***
occupationother         -1.270e-02  1.123e-02   -1.131 0.257974    
occupationprogrammer     3.925e-03  1.232e-02    0.319 0.749996    
occupationretired       -7.613e-02  2.121e-02   -3.590 0.000331 ***
occupationsalesman      -1.783e-02  2.639e-02   -0.675 0.499461    
occupationscientist     -5.003e-02  1.845e-02   -2.712 0.006679 ** 
occupationstudent        2.686e-02  1.090e-02    2.463 0.013767 *  
occupationtechnician    -5.870e-02  1.523e-02   -3.854 0.000116 ***
occupationwriter         2.644e-02  1.318e-02    2.005 0.044929 *  
Action                  -5.722e-02  7.492e-03   -7.638 2.22e-14 ***
Adventure                1.745e-02  8.584e-03    2.032 0.042109 *  
Animation                4.274e-01  1.688e-02   25.323  < 2e-16 ***
Childrens               -2.551e-01  1.249e-02  -20.415  < 2e-16 ***
Comedy                  -4.746e-02  7.045e-03   -6.736 1.63e-11 ***
Crime                    1.119e-01  9.423e-03   11.877  < 2e-16 ***
Documentary              4.487e-01  2.806e-02   15.989  < 2e-16 ***
Drama                    2.757e-01  6.877e-03   40.087  < 2e-16 ***
Fantasy                 -1.704e-01  2.173e-02   -7.841 4.51e-15 ***
Film.Noir                1.904e-01  2.088e-02    9.121  < 2e-16 ***
Horror                  -1.998e-01  1.178e-02  -16.965  < 2e-16 ***
Musical                 -1.412e-01  1.390e-02  -10.158  < 2e-16 ***
Mystery                  7.144e-02  1.206e-02    5.926 3.10e-09 ***
Romance                  1.055e-01  6.493e-03   16.255  < 2e-16 ***
Sci_Fi                   4.949e-02  8.492e-03    5.827 5.65e-09 ***
Thriller                 8.037e-02  7.077e-03   11.358  < 2e-16 ***
War                      1.514e-01  9.222e-03   16.422  < 2e-16 ***
```

5. Example for cosine similarity recommender

1) Calculate the cosine similarity between user 10 and every other user in the MovieLens user network
2) Based on cosine similarity, the five most similar users for user 10 are user 202, user 1028, user 181, user 48 and user 318.
3) Identify the five top ranked movies that have not been watched by user 10, based on ratings from user 202, user 1028, user 181, user 48 and user 318:

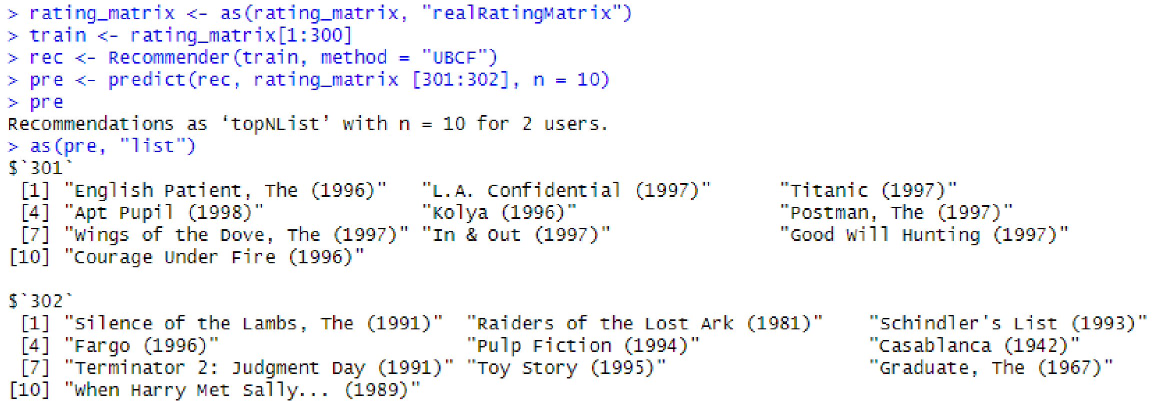
Movie title/User	202	1028	181	48	318	Mean
Toy Story (1995)	4.0	5.0	5.0	5.0	5.0	4.75
Star Wars (1997)	4.0	5.0	5.0	5.0	5.0	4.75
Lion King (1994)	5.0	4.0	5.0	5.0	5.0	4.75
Alien (1979)	5.0	4.0	5.0	5.0	5.0	4.75
Scream (1996)	4.0	4.0	5.0	5.0	5.0	4.6

The five top ranked movies: Toy Story, Star Wars, Lion King, Alien and Scream will be recommended to user 10.

6. Sample code for user-based collaborative filtering with "recommenderlab"

Collaborative filtering (CF) uses given rating data by many users for many items as the basis for predicting missing ratings and/or for creating a top-N recommendation list for a given user, called the active user. From the description of the "recommenderlab" package, we learned that the basic assumption of user-based collaborative filtering is that users with similar preferences will rate items similarly. Thus, missing ratings for a user can be predicted by first finding a neighborhood of similar users and then aggregate the ratings of these users to form a prediction.

The UBCF function from "recommenderlab" package is convenient to use. In our dataset, the rating_matrix has 943 user_id on the row and 1682 movie_title on the column, with corresponding rating from each user for each movie as input. After transforming the matrix into the default format of function Recommender(), a movie recommender was built using the first 300 users. And the recommender was used to create top-10 recommendation movie lists for user 301 and user 302.



```
> rating_matrix <- as(rating_matrix, "realRatingMatrix")
> train <- rating_matrix[1:300]
> rec <- Recommender(train, method = "UBCF")
> pre <- predict(rec, rating_matrix [301:302], n = 10)
> pre
Recommendations as ‘topNList’ with n = 10 for 2 users.
> as(pre, "list")
$`301`
 [1] "English Patient, The (1996)"   "L.A. Confidential (1997)"      "Titanic (1997)"
 [4] "Apt Pupil (1998)"              "Kolya (1996)"                  "Postman, The (1997)"
 [7] "Wings of the Dove, The (1997)" "In & Out (1997)"               "Good Will Hunting (1997)"
[10] "Courage Under Fire (1996)"

$`302`
 [1] "Silence of the Lambs, The (1991)"  "Raiders of the Lost Ark (1981)"    "Schindler's List (1993)"
 [4] "Fargo (1996)"                      "Pulp Fiction (1994)"               "Casablanca (1942)"
 [7] "Terminator 2: Judgment Day (1991)" "Toy Story (1995)"                  "Graduate, The (1967)"
[10] "When Harry Met Sally... (1989)"
```