EE232E Project 4

IMDb Mining

Spring 2018

Yuting Tang, 304881011

Fangjia Zhu, 905036438

Jiahui Li, 004356402

Xiaokun Zhang, 104775945

I. Introduction

In this project, we studied the directed actor/actress network and undirected movie network induced from the Internet Movie Database (IMDb). For the actor/actress network, properties such as in-degree distribution, actor pairs and top actors/actresses were obtained. For the movie network, the community and neighborhood structures were obtained and the movie ratings were predicted by both regression and bipartite graph methods.

II. Analysis

1 Actor/Actress network

In this part of the project, we have done some preprocessing on the two text files about actors and actresses. First of all, we need to merge the two text files into one and remove the actor/actress who has acted in less than 10 movies. Then we need to clean the merged text file since there might be same movie counted multiple times due to the different roles of the actor/actress in the movie. After the preprocessing is done, the conclusion is as follows:

- total number of actors: **74598**
- total number of actresses: **38534**
- total number of actors and actresses: **113132**
- total number of unique movies: **468149**

1.1 Directed actor/actress network creation

In this part, we have used the processed text file to create the directed actor/actress network. The nodes of the network are the actor/actress we have obtained in the last part and the weighted edges are computed by $w_{i\to j} = \frac{|S_i \cap S_j|}{|S_i|}$. The edge-list is saved as "mergeNew.txt". After creating the weighted directed actor/actress network, we have plot the in-degree distribution of the actor/actress network, which is shown in the following figure.

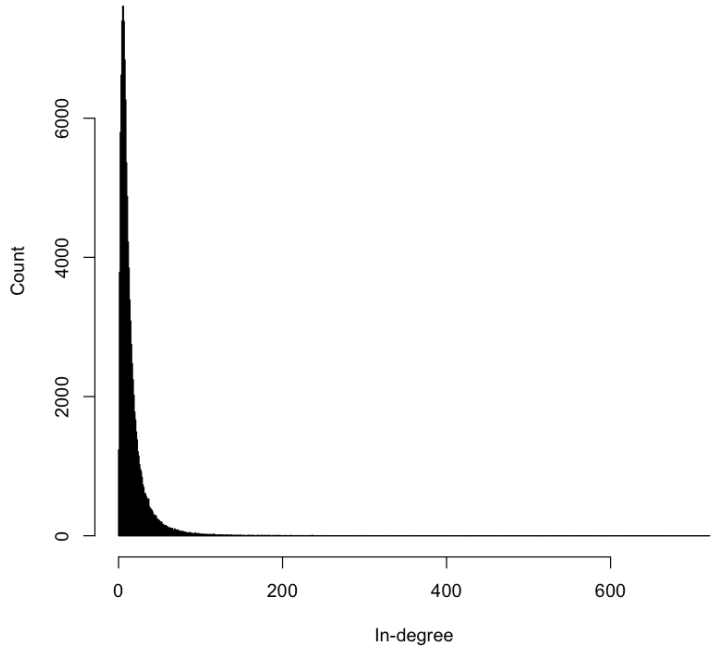

Figure. 1. The in-degree distribution of the actor/actress network

As you can see from the figure above, the in-degree distribution is heavily right-skewed with the most density concentrated in the region of degree less than 100. Since the network was built based on weights between actors/actresses casting in the same movie, we can imagine that it would be quite hard for an actor or actress to be involved with more than 100 movies for a career lasting for at most several decades. From the distribution however, there are still actor/actress who has appeared in more than 600 movies.

1.2 Actor pairings

In this section, we have tried to find the pairing between actors, considering the 10 actors provided in the statement. For each of the actors listed, we firstly calculate the edge weight between the input actor and every other actor and find the biggest one, which is exactly the output actor that the input actor prefers to work the most. Notice that the expression of actors' name in the processed text file is (Last-name, First-name), while the list of actors is shown as (First-name Last-name). The result is shown in the following table.

Input Actor	Output Actor	Edge Weight
Tom Cruise	Nicole Kidman	0.1746031746031746
Emma Watson (II)	Daniel Radcliffe	0.52
George Clooney	Matt Damon	0.11940298507462686
Tom Hanks	Tim Allen (I)	0.10126582278481013
Dwayne Johnson (I)	Steve Austin (IV)	0.20512820512820512
Johnny Depp	Helena Bonham Carter	0.08163265306122448
Will Smith (I)	Darrell Foster	0.12244897959183673
Meryl Streep	Robert De Niro	0.061855670103092786
Leonardo DiCaprio	Martin Scorsese	0.10204081632653061
Brad Pitt	George Clooney	0.09859154929577464

Table 1. The Actor Pairings with Edge Weight

Most of the actor pairs listed above indeed revealed the relationships underlying between actors. For example, Nicole Kidman had a marriage with Tom Cruise and Emma Watson and Daniel Radcliffe are famous because of the Harry Potter series. However, for some particular actor pairs the result returned does not necessarily make sense. If carefully examine the edge weights involved with Meryl Streep, we can see that that there are multiple output actors with the same weight of 0.062, as a result Robert De Niro would not necessarily be the most prefered actor for Meryl Streep, although he would definitely be among the prefered ones.

1.3 Actor rankings

In this section, we have extracted the top 10 actor/actress from the network through the google's pagerank algorithm. After finding the top 10 actress/actress in the network, we have also reported the number of movies that the actor/actress is in as well as the in-degree of each one in the top 10 list. What is more, we need to compare this top 10 list with the list provided in the last section. The result is as follows.

Top 10 Actor Name	Number of Movies	In-degree	PageRank
Bess Flowers	17	720.1888	0.0002351439
Fred Tatasciore	71	347.0015	0.0001989839
Sam Harris (II)	11	620.8671	0.0001972129
Steve Blum (IX)	74	334.0308	0.0001955247
Harold Miller (I)	18	534.7349	0.0001727242
Ron Jeremy	20	237.4834	0.0001584708
Lee Phelps (I)	18	397.5734	0.0001573222
Yuri Lowenthal	14	277.4578	0.0001567747
Robin Atkin Downes	18	264.2039	0.0001517972
Frank O'Connor (II)	15	377.3061	0.0001469572

Table 2. The Top 10 Actor/Actress with Movie Number and In-degree

According to the result, the top 10 list **does not** have any actor/actress listed in the previous section. One possible reason for this phenomenon is that the pagerank algorithm used here considers only the visiting probability of the nodes with random walk. As a result, an actor with more edges - and thus larger total edge weights and larger in-degree, would be assigned a larger pagerank score by the algorithm.

Then to justify this argument, we have reported the pagerank scores of the actor/actress listed in the previous section. In addition, we have also reported the number of movies each of these actor/actress have acted in as well as their in-degree so that comparisons can be made. As you can see from Table 2 and 3, even though the actors mentioned in Section 1.2 are indeed famous and involved in much more movies than those in Table 2, their in-degrees are nearly 10 times smaller, and correspondingly their pagerank scores are much smaller.

Input Actor	Number of Movies	In-degree	PageRank
Tom Cruise	63	79.72513	2.11702e-05
Emma Watson (II)	71	31.86726	3.01397e-06
George Clooney	67	77.50288	6.741416e-06
Tom Hanks	79	103.9414	4.116723e-06
Dwayne Johnson (I)	78	94.52225	4.280561e-05
Johnny Depp	98	103.8478	5.68508e-06
Will Smith (I)	49	68.77712	3.580536e-05
Meryl Streep	97	75.0736	3.831985e-06
Leonardo DiCaprio	49	62.78713	2.830433e-06
Brad Pitt	71	83.76134	2.517639e-05

Table 3. The Listed 10 Actor/Actress with Movie Number and In-degree

2 Movie network

2.1 Undirected movie network creation

In this section, we have created an undirected movie network and then explored the various structural properties of the network. The processed text files are the same as those from the previous section to create the movie network. However, the nodes of this network are the movies and the weights of the edges are given by $w_{ij} = \frac{|A_i \cap A_j|}{|A_i \cup A_j|}$. The edge-list is saved as "movie_edge_list2.txt". After creating the weighted undirected movie network, we have plot the degree distribution of the movie network, which is shown in the following figure.

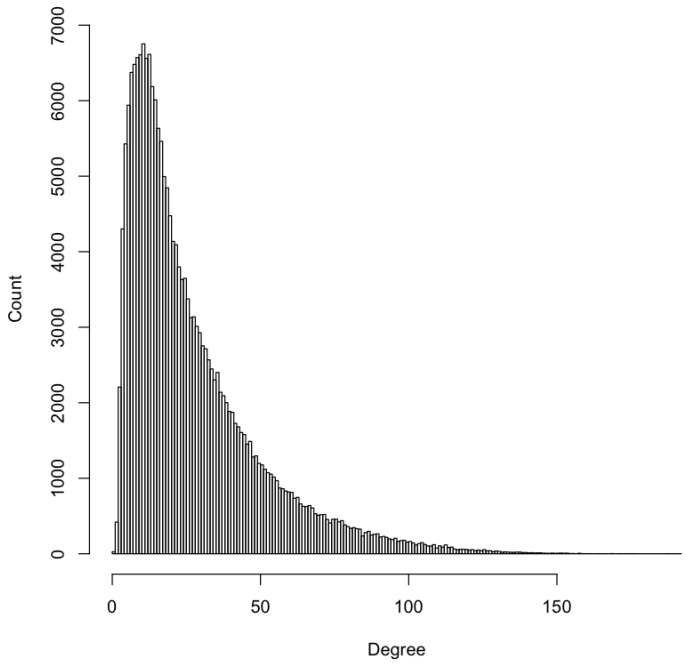

Figure. 2. The degree distribution of the movie network

The movie network degree distribution is also heavily right-skewed where most of the nodes are having a degree of less than 50. In other words, most of the movies are having less than 50 actors performing in them if assuming one actor would perform in any movie. However, in reality often times actors would usually prefer to work with some certain actors, directors, or directors and this fact might also contribution to such distribution of movie network.

2.2 Communities in the movie network

In this part, we have extracted the communities in the movie network and explored their relationship with the movie genre. Firstly, we need to load the “movie_genre.txt” file. Then we have found the communities in the movie network by the Fastgreedy community detection algorithm. Using the Fast Greedy community detection algorithm, we have created “moviefg.rds” file to save the data of communities so that we could use it more conveniently in the later parts.

The sizes of the communities in the movie network are listed in the following table.

Community Number	1	2	3	4	5	6	7	8	9
Community Size	45444	35025	27082	5215	13101	8500	178	7282	20328
Community Number	10	11	12	13	14	15	16	17	18
Community Size	1495	13156	4296	740	1115	4829	1706	827	4101
Community Number	19	20	21	22	23	24	25	26	27
Community Size	676	14	2075	5917	17	11	18	402	22

Table 4. The Community Sizes of the Movie Network

Then, we picked the biggest 9 communities and the 20th community (The size is 14, which satisfies the request of Question 8(c).) as the 10 communities (1, 2, 3, 9, 11, 5, 6, 8, 22, 20). And for each community we need to plot the distribution of the genres of the movies in the community. What we need to do is to find out the genre of each movie, and count the number of each genre for each community.

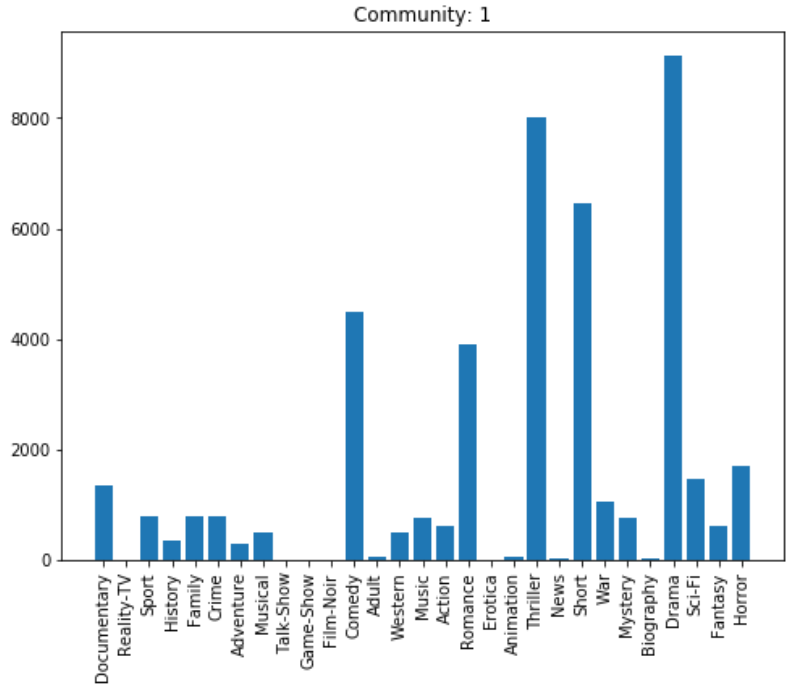

Figure. 3. The distribution of the genres of the movies in Community 1

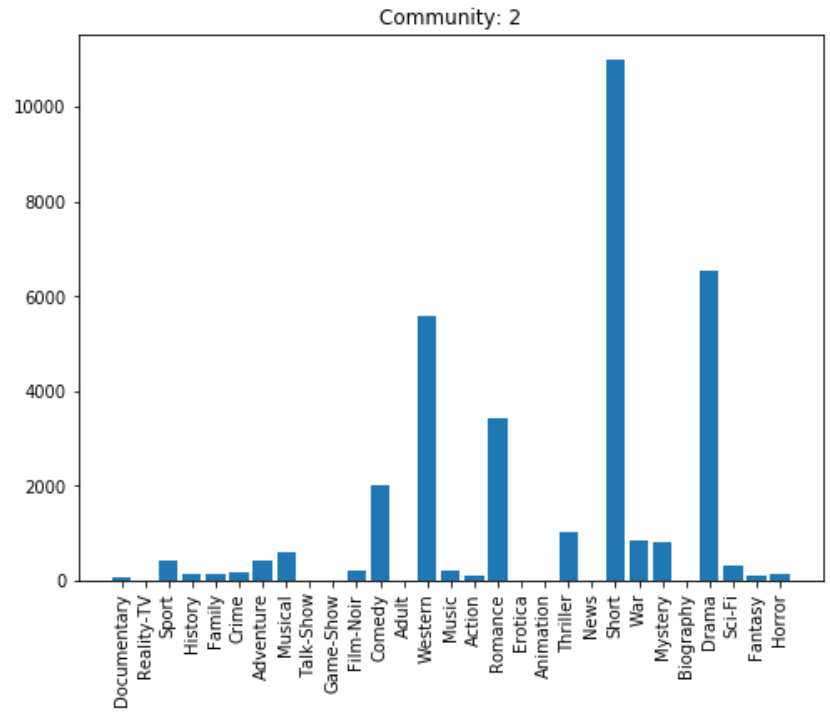

Figure. 4. The distribution of the genres of the movies in Community 2

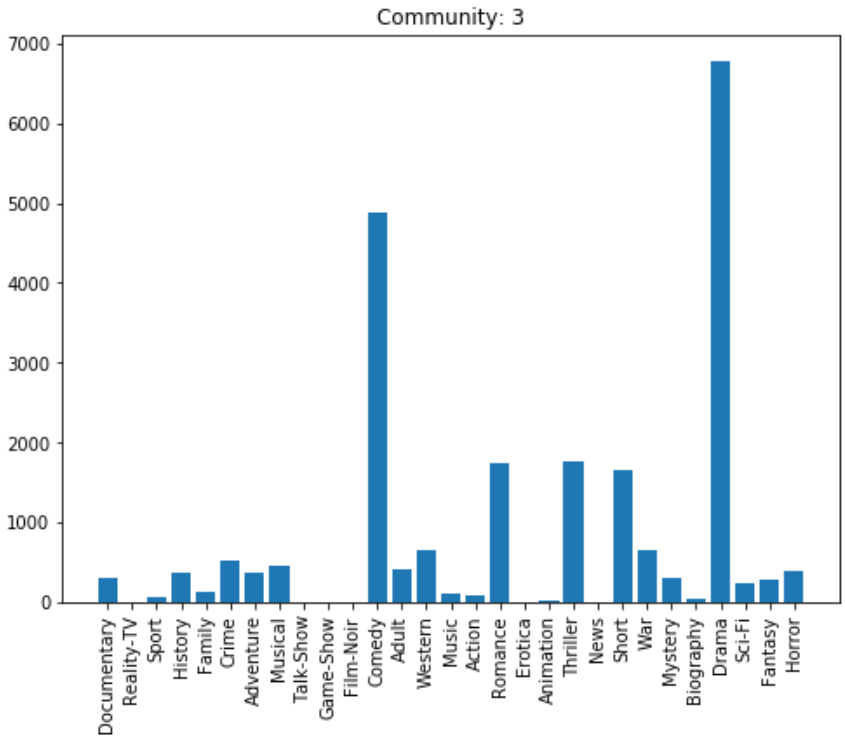

Figure. 5. The distribution of the genres of the movies in Community 3

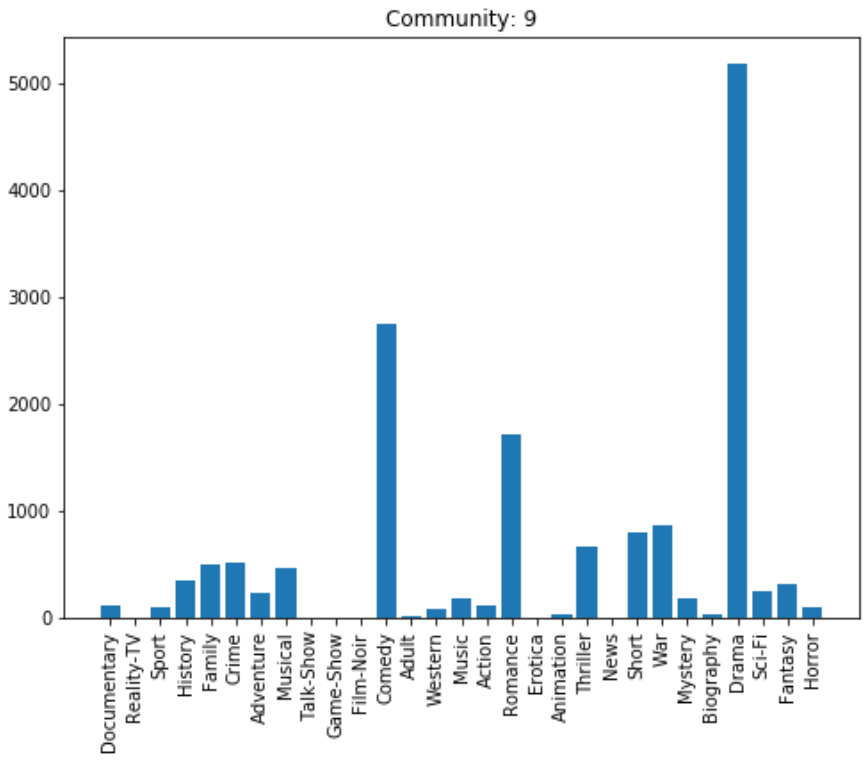

Figure. 6. The distribution of the genres of the movies in Community 9

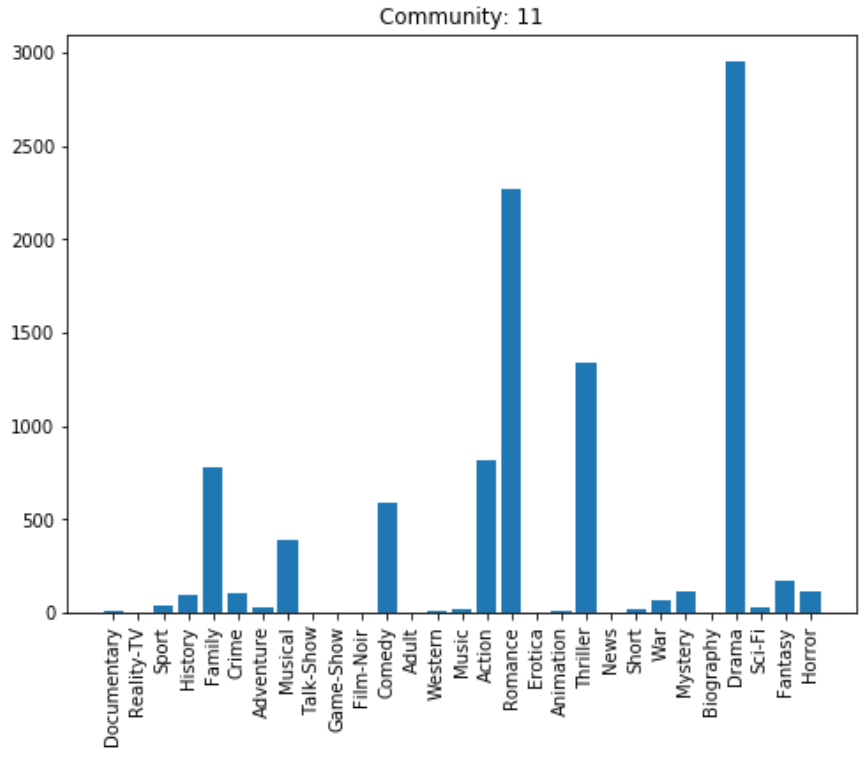

Figure. 7. The distribution of the genres of the movies in Community 11

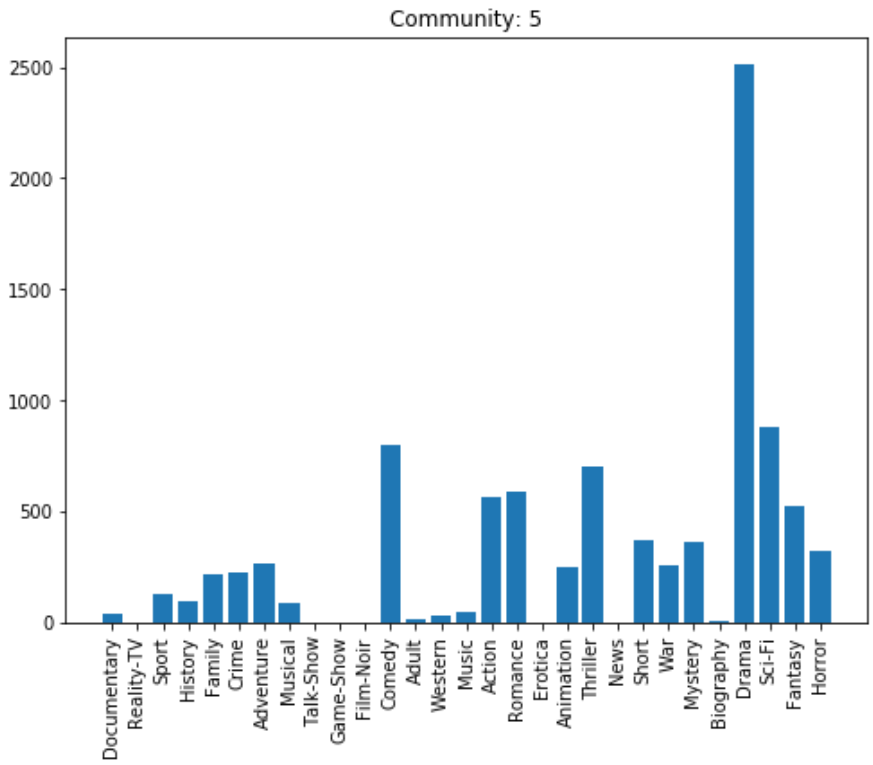

Figure. 8. The distribution of the genres of the movies in Community 5

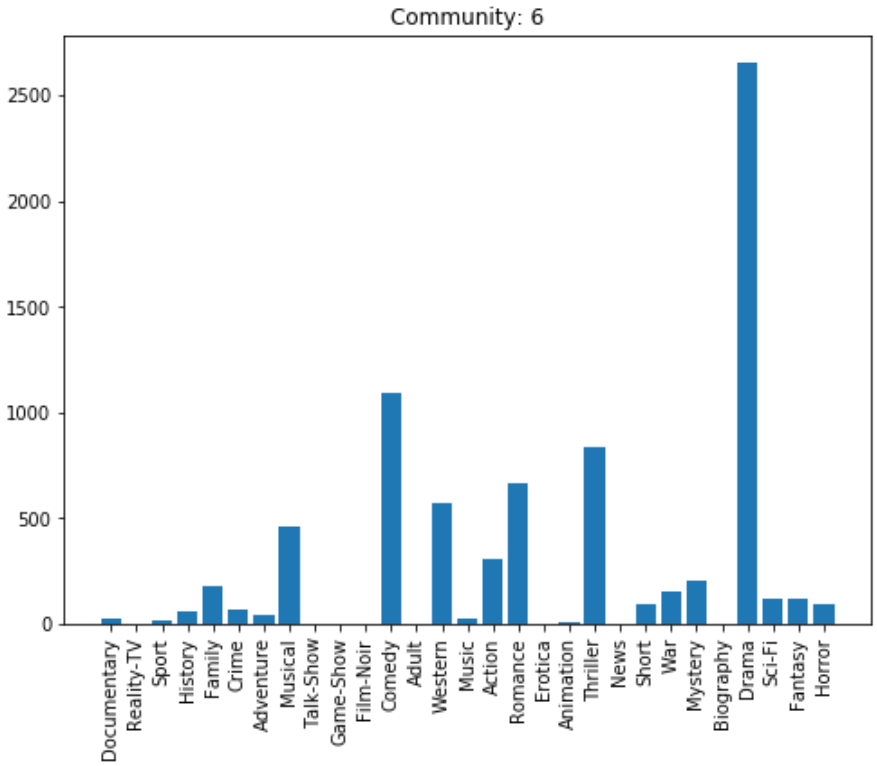

Figure. 9. The distribution of the genres of the movies in Community 6

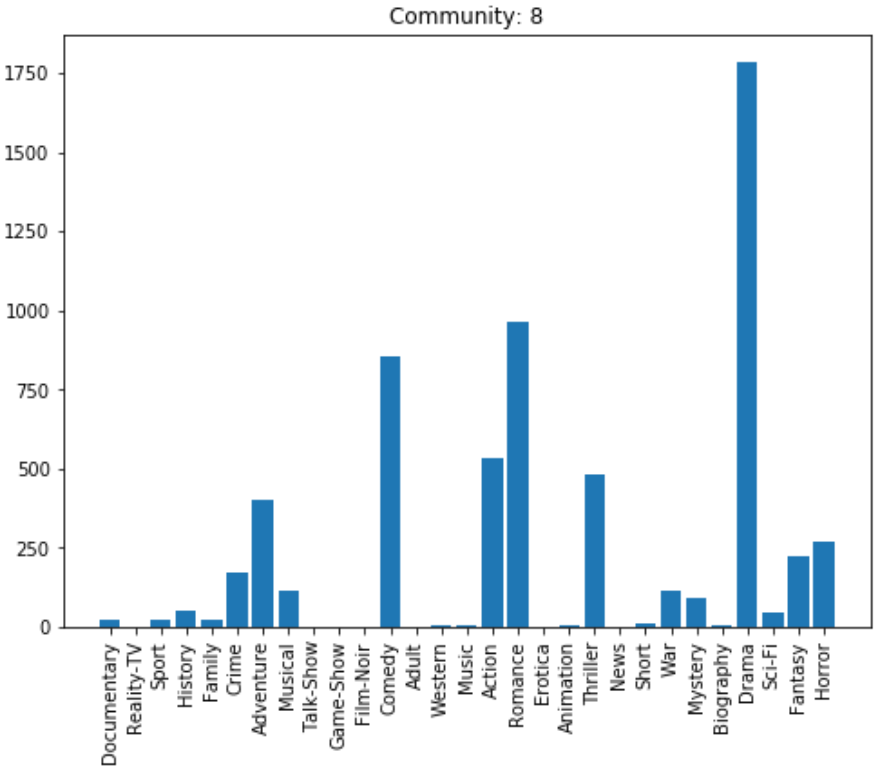

Figure. 10. The distribution of the genres of the movies in Community 8

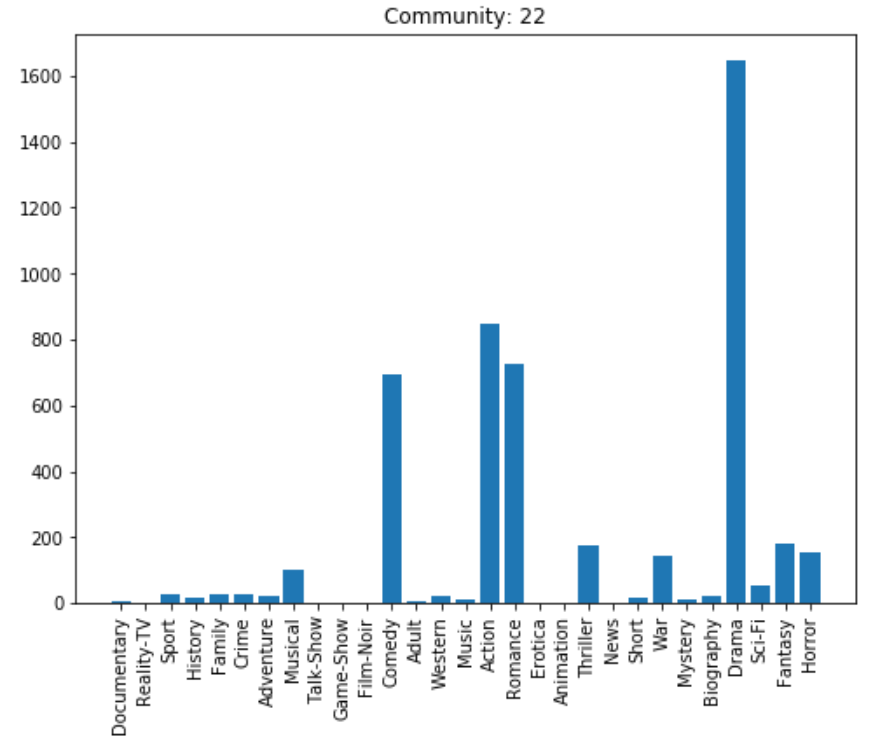

Figure. 11. The distribution of the genres of the movies in Community 22

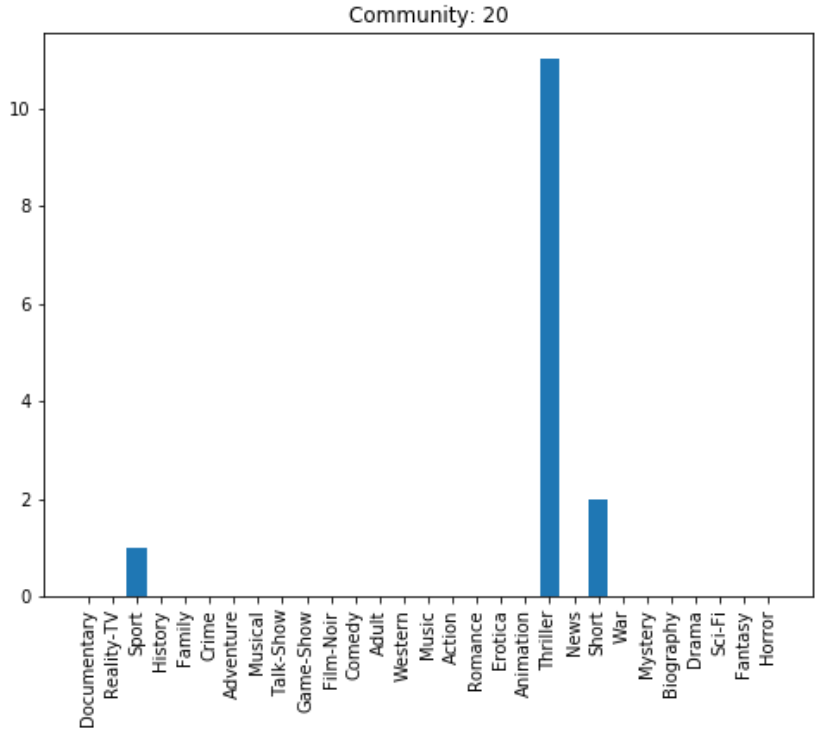

Figure. 12. The distribution of the genres of the movies in Community 20

In this part, we need to determine the most dominant genre based simply on frequency counts in each community. The result is shown in the following table.

Community Number	**1**	**2**	**3**	**9**	**11**	**5**	**6**	**8**	**22**	**20**
Dominant genre	Drama	Short	Drama	Drama	Drama	Drama	Drama	Drama	Drama	Thriller

Table 5. The Dominant Genre Based on Frequency Counts

As we could see, the most frequent dominant genre across communities based on frequency counts is Drama. One possible explanation is that drama is only opposite to comedy and many other genres can actually be classified as drama, for example horror and romance as a broad genre can also be considered as horror drama and romantic drama. Since movies in the dataset can at most have one label of genre, while in the actual IMDb database movies would have

several genres labeled, most movies that are not comedy or tragedy would naturally be more likely to be labeled as drama. Another possible explanation is that the film industry are most interested in drama for some unknown reason and making this particular genre most popular in the database.

In this part we have calculated the score of each genre in each community by $ln(c(i)) * \frac{p(i)}{q(i)}$, where c(i) is the number of movies belonging to genre i in the community, p(i)is the fraction of genre i movies in the community, and q(i) is the fraction of genre i movies in the entire data set. And the most dominant genre in each community based on the modified scores is shown below.

Community Number	**1**	**2**	**3**	**9**	**11**	**5**	**6**	**8**	**22**	**20**
Dominant genre	Erotica	Talk-Show	Reality-TV	Talk-Show	Talk-Show	Talk-Show	Talk-Show	Talk-Show	Talk-Show	History

Table 6. The Dominant Genre Based on Modified Scores

As we could see, the most dominant genre in each community based on the modified scores is Talk-Show. By looking at the formulation of the score, the term p(i)/q(i) would be used for normalization, removing the impact of the most popular genre across all communities and obtaining the "true" most popular genre in the community. The term ln(c(i)) could be used to transform the number of movies belonging to each genre and make them comparable across the communities. Intuitively, by applying such score, the dominant genre in each community would be different and would not necessarily be Drama anymore. However, it is somewhat surprising to see the result in Table 6, where almost all the communities are dominated by talk shows. This finding would probably result from either the choice of communities, the problem of mislabeling genres mentioned above, or the formulation of the score itself.

As mentioned before, the community of movies that has size between 10 and 20 chosen by us is Community 20. In this part, we have determined all the actors who acted in these movies and

plot the corresponding bipartite graph (i.e. restricted to these particular movies and actors). The graph is shown below (Movies are plot as red nodes, while actors are plot as blue nodes).

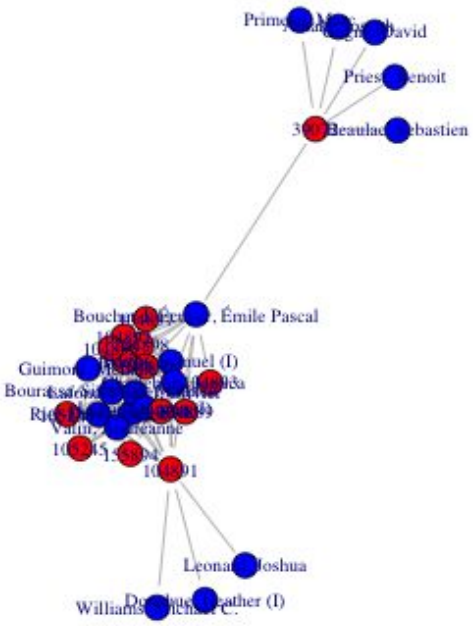

Figure. 13. The Bipartite Graph of Movies and Actors

In order to determine three most important actors, we just count the number of movies that the actor/actress acted in and find out three most frequent actors. The result is **Olivier Lafond-Martel, Simon Legros (I), Nick Desjardins** who all acted **13** movies and the total number of movies in the community is **14**. Basically, these three actors helped forming the community by acting in almost all the movies in the community, such that at least 3 edges between each node to each other node can be formed, and as a result, the modularity score (which is used by fast greedy algorithm to determine the community structure) would be relatively high. The correlation between these actors and the dominant genre of this community (Thriller from Table 5) is also quite obvious: by browsing the IMDb database online, we found that all the movies acted by these 3 actors indeed have the genre tag Thriller in common. However, the correlation between the actors and the dominant genre found for this community in Table 6 (History) remains unclear at the time of writing this report.

2.3 Neighborhood analysis of movies

In this part of the project, we've explored the neighborhood of the following three movies using the `movie_rating.txt`:

• Batman v Superman: Dawn of Justice (2016)

• Mission: Impossible - Rogue Nation (2015)

• Minions (2015)

First for each of the listed movie, we extracted its neighbors and plotted the rating distribution of the available movies in the neighborhood, as shown in Figures 14 to 16, respectively. The results of actual rating of the movie, average rating of its neighbors and most frequent rating in neighborhood were summarized and compared in Table 7. As for *Batman v Superman: Dawn of Justice (2016)*, the movie id was extracted as 10321 with the average rating of neighbors as 5.79 and the most frequent rating in neighborhood as 6.4 with 38 instances. As for *Mission: Impossible - Rogue Nation (2015)*, the movie id was extracted as 39182 with the average rating of neighbors as 5.64 and the most frequent rating in neighborhood as 6.2 with 31 instances. As for *Minions (2015)*, the movie id was extracted as 78995 with the average rating of neighbors as 6.07 and the most frequent rating in neighborhood as 6.8 with 28 instances.

From Table 7, we could see that the value of the average rating in neighborhood for each movie is generally lower than that of the rating of the movies whose neighbors have been extracted, while the values of the most frequent rating in neighborhood are generally closer to that of the actual ratings.

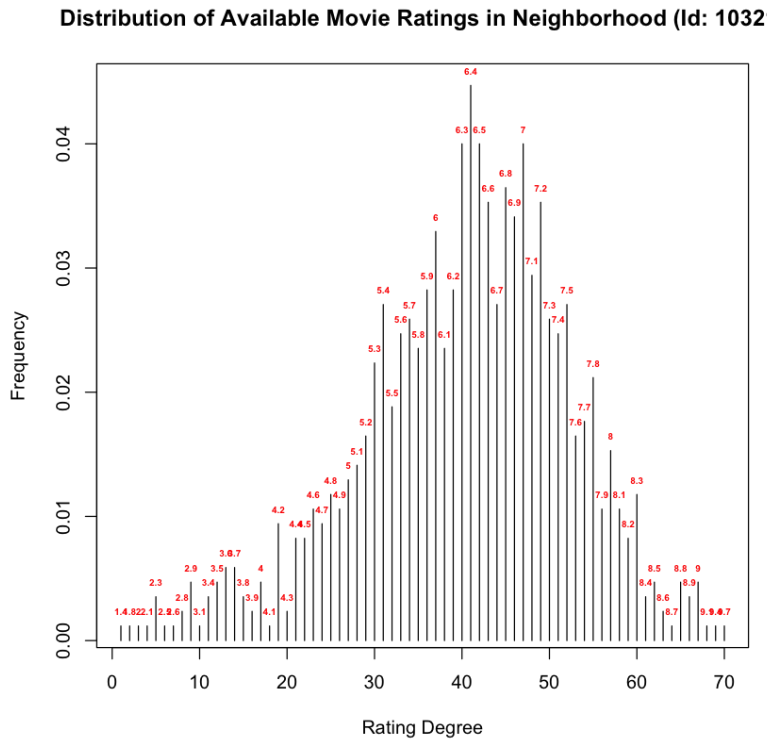

Figure. 14. available movie rating distribution in the neighborhood for *Batman v Superman*

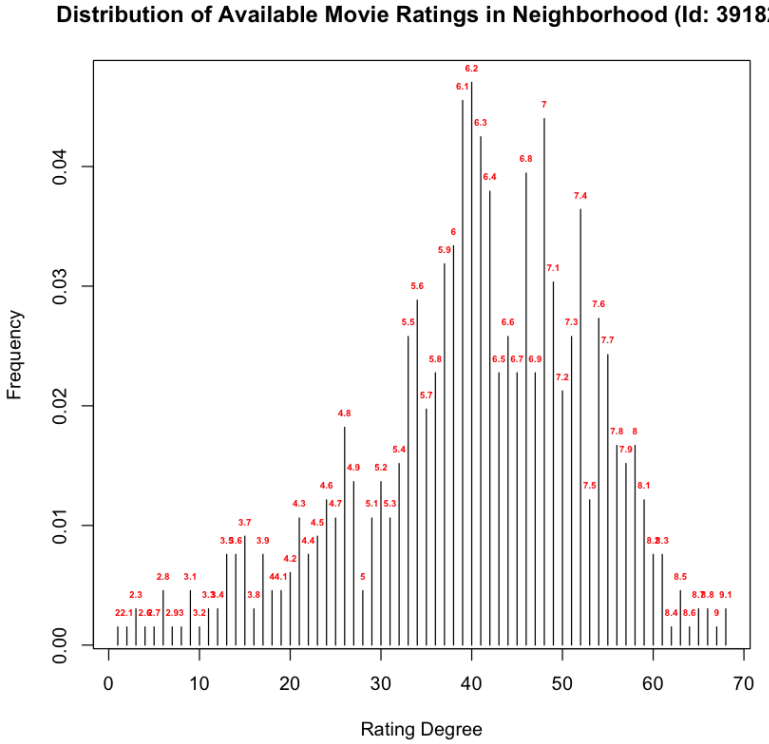

Figure. 15. available movie rating distribution in the neighborhood for *Mission: Impossible*

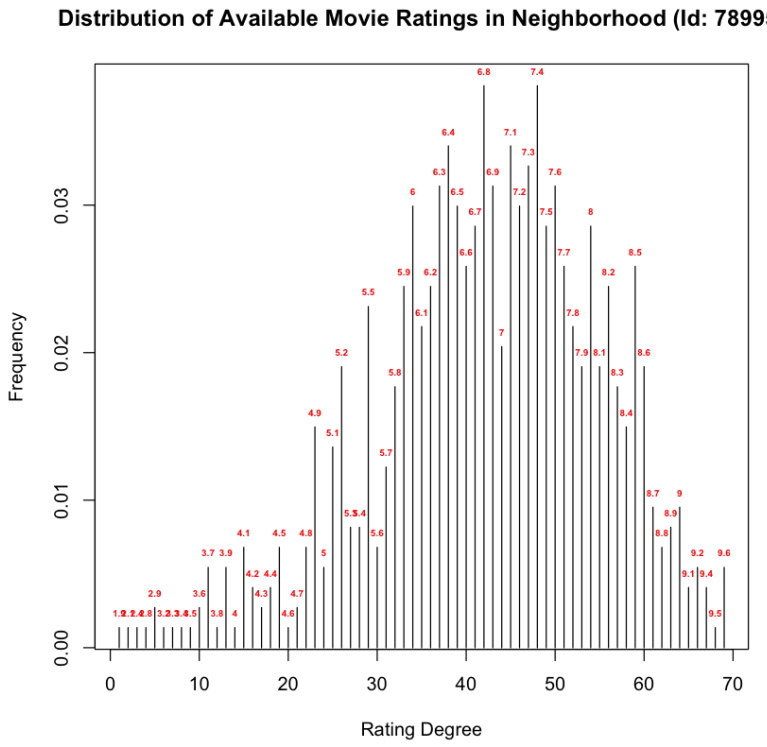

Figure. 16. available movie rating distribution in the neighborhood for *Minions*

Movie Name	Movie ID	Movie Rating	Average Rating in Neighborhood	Most Frequent Rating in Neighborhood
Batman v Superman: Dawn of Justice (2016)	10321	6.6	5.79	6.4
Mission: Impossible - Rogue Nation (2015)	39182	7.4	5.64	6.2
Minions (2015)	78995	6.4	6.07	6.8

Table 7. Rating Summary for the Selected Movies

Now, we repeated the previous steps but limited the neighborhood to consist of movies from the same community, and again for each movie, we extracted its neighbors and plotted the rating distribution of the available movies in the neighborhood, as shown in Figures 17 to 19,

respectively. The results of actual rating of the movie, average rating of its neighbors and most frequent rating in neighborhood were summarized and compared in Table 8. As for *Batman v Superman: Dawn of Justice (2016)*, the average rating of neighbors became 5.69, and the most frequent rating in neighborhood became 6.4 with 34 instances. As for *Mission: Impossible - Rogue Nation (2015)*, the average rating of neighbors became 5.74, and the most frequent rating in neighborhood became 6.1 with 26 instances. As for *Minions (2015)*, the average rating of neighbors became 6.45, and the most frequent rating in neighborhood became 7.3 with 19 instances.

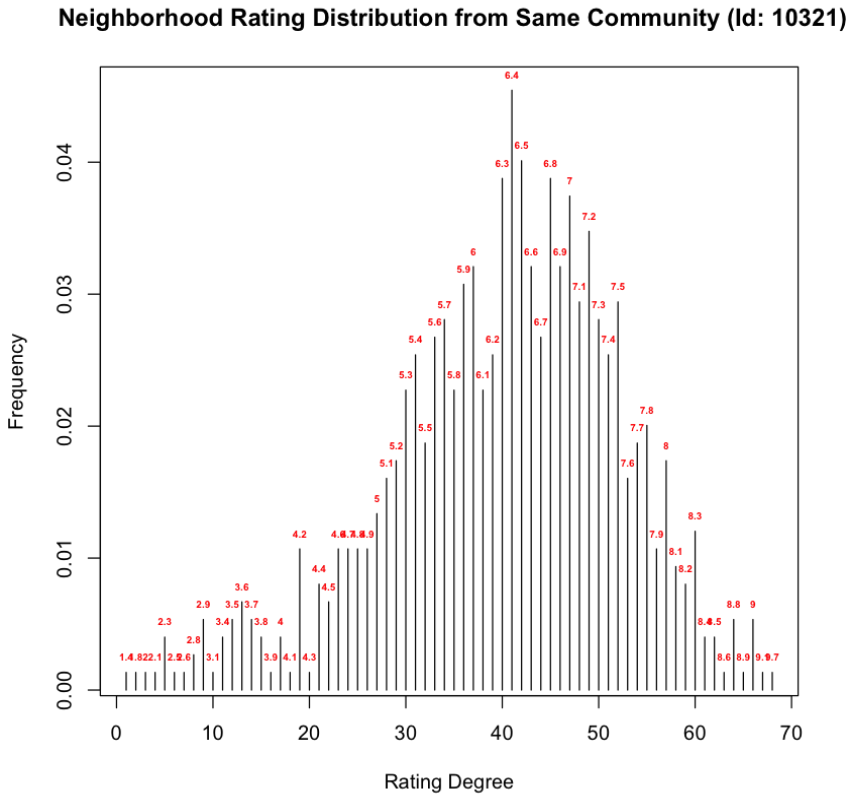

Figure 17. Neighborhood rating distribution from the same community for *Batman v Superman*

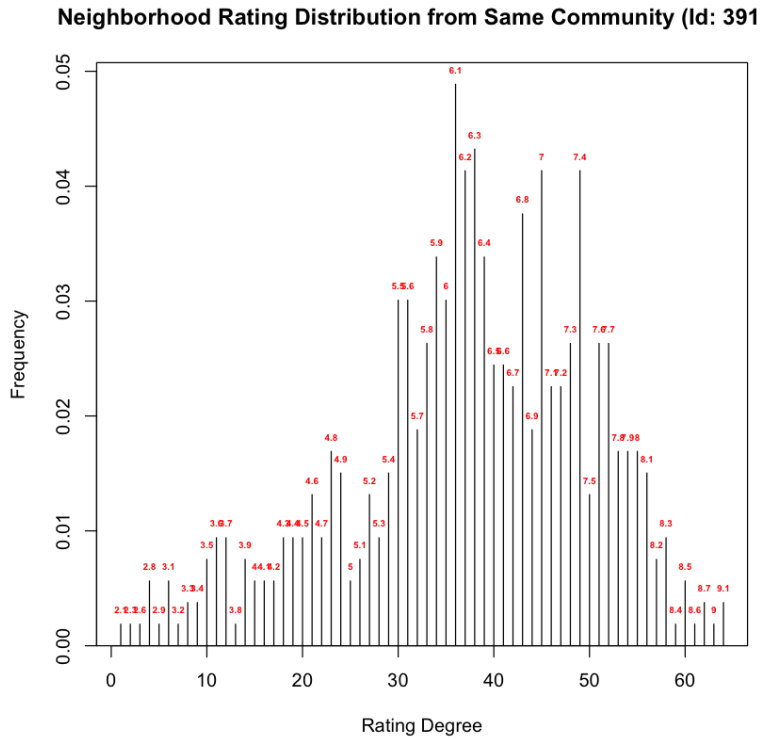

Figure 18. Neighborhood rating distribution from the same community for *Mission: Impossible*

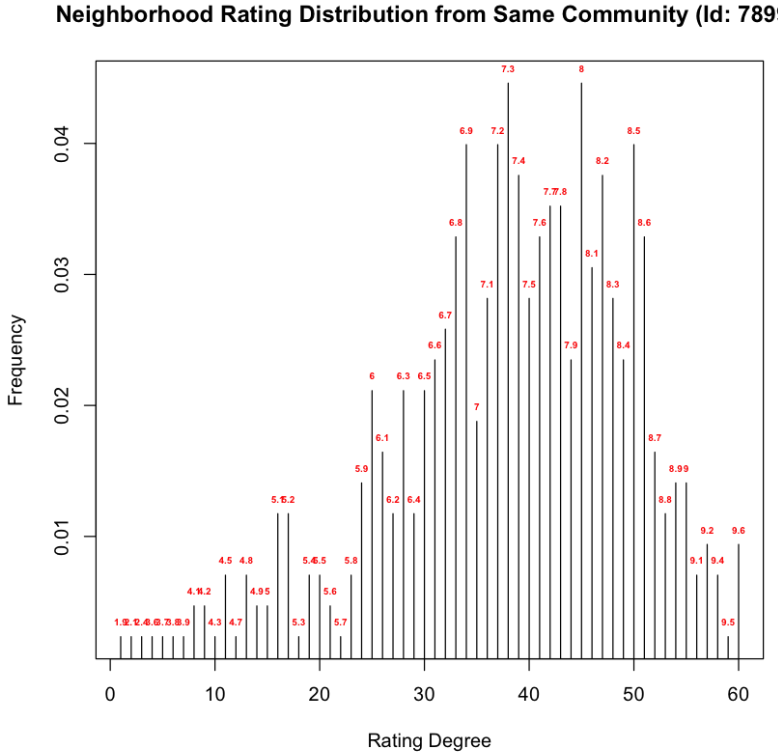

Figure 19. Neighborhood rating distribution from the same community for *Minions*

Movie Name	Movie ID	Movie Rating	Average Rating in Neighborhood	Most Frequent Rating in Neighborhood
Batman v Superman: Dawn of Justice (2016)	10321	6.6	5.69	6.4
Mission: Impossible - Rogue Nation (2015)	39182	7.4	5.74	6.1
Minions (2015)	78995	6.4	6.45	7.3

Table 8. Rating Summary for Selected Movies from the Same Community

We could see from Table 8 that after restricting the neighborhood to consist of movies from the same community, there is a better match between the average rating of the movies in the restricted neighborhood and the rating of the movie whose neighbors have been extracted. Although, the value of the average rating in neighborhood for each movie is still generally lower than that of the rating of the movies whose neighbors have been extracted, the values of the most frequent rating in neighborhood become much generally closer to that of the actual ratings.

Finally, we extracted the top five neighbors for each of the listed movie and sorted them based on the edge weights in a descending order. The top five neighbors for each movie and their associated community membership are summarized in Table 9, in which the order of the neighbors edge weight for each movie is listed from right to left in a descending manner.

Batman v Superman: Dawn of Justice (2016)	**Neighbor IDs**	22165	10363	33301	9502	3384
	Community Membership	1	1	1	1	1
Mission: Impossible - Rogue Nation (2015)	**Neighbor IDs**	32741	32744	57762	68813	39183
	Community Membership	11	11	1	1	1

Minions (2015)	**Neighbor IDs**	37617	16741	37589	52491	61332
	Community Membership	5	5	5	5	5

Table 9. Summary for Top Five Neighbor IDs and Associated Community Membership in Descending Order

2.4 Predicting ratings of movies

Regression Model

In this section we determined the rating of the following movies using linear regression model:

• Batman v Superman: Dawn of Justice (2016)

• Mission: Impossible - Rogue Nation (2015)

• Minions (2015)

The following features are used to make the training and prediction. We used pageranks that are calculated previously and picked the top 5 scores among the actors/actresses involved in the movie by removing movies which contains less than 10 actors/actresses and actor/actress participates in less than 10 movies. We ranked the movies in the movie_rating.txt in descending order, and picked the first 100 directors' names involved in the high score movies. The first 5 items are 5 highest page ranks, the sixth item is a boolean value of whether the director is in the top 100 directors list. If so, the remaining items show his/her ranking. The corresponding item will be 1 according to the ranking, if not, the value will be 0 And finally, we calculated root mean squared error(RMSE) to evaluate the predicting results.

Movie Name	Actual Rating	Predicted Rating
Batman v Superman: Dawn of Justice (2016)	6.6	6.21
Mission: Impossible - Rogue Nation (2015)	7.4	6.17
Minions (2015)	6.4	6.18

Table 10. Actual and Predicted Ratings for Selected Movies

The RMSE can represent the sample standard deviation of the differences between predicted values and observed values. The RMSE we got is **0.956** for the regression model. The predicted ratings are acceptable compared to the actual rating, but it's not accurate. The reason for that is we only have a limited number of parameters, to be more specific, the feature space that we considered is not directly correlated to the rating of these movies. Because only the supporting actors who have acted in many movies have a high PageRank compared to lead or star actors. In other words, the rating of a movie should determined by the main leading actor rather than support actors.Perhaps instead of looking at the PageRank of the actors in the movie, if we could have a given weighted measure to the role played by the actors and then taken the top 5 values from this measurements, we will have a better results. So the better feature space we have, the more precise predictions we will have.

Bipartite Graph Approach

In this section we tried to predict the movie ratings from a bipartite graph between movies and actors. A bipartite graph is a graph whose vertices can be divided into two disjoint sets U and V where U and V are each independent sets such that every edge connects a vertex in U to one in V. V1 represents the set of actor/actresses, V2 represents the set of movies, the edges are created between the actor/actress and each movie they participated. In each individual subset, there are no connections, which means there are no edge between the movies and there are no edges between actors or actress. The features we used are the same as before, which is removing all the actor/actress participated in less than 10 movies.

Firstly, we assign weight to each of them mapping every movie to corresponding actor as given in the actor_movie.txt. Then, the bipartite graph can be completely established when we use the information presented in previous step such a way that for each movie. Finally, we can calculate the average rating of all its actors present in the movies.

Movie Name	Actual Rating	Predicted Rating
Batman v Superman: Dawn of Justice (2016)	6.6	6.41
Mission: Impossible - Rogue Nation (2015)	7.4	6.83
Minions (2015)	6.4	6.24

Table 11. Actual and Predicted Ratings for Selected Movies using Bipartite Graph Approach

From Table 11, it can be seen that this algorithm yields better results than the regression model algorithm. The RMSE is **0.7921,** this algorithm performs better because it uses actor weight to predict movie ratings. The actor weight mechanism filters out the irrelevant information from the prediction mechanism and as a result the predictions are better. However, the possible reason for the slightly high error rate in Mission : Impossible is their average ratings may not consistent in this computation. But overall, bipartite graph mechanism is better than regression model in Q12.