ECE 232E: Large-Scale Social and Complex Networks: Models and Algorithms

Project 4: IMDb Mining

Akshay Sharma (504946035)

Anoosha Sagar (605028604)

V*c*Nikhil Thakur(804946345)

Rahul Dhavalikar (205024839)

1. Actor/Actress network

1) Perform the preprocessing on the two text files and report the total number of actors and actresses and total number of unique movies that these actors and actresses have acted in.

Ans:

In this project, we are dealing with actor, actress, movie, genre, and rating information derived from the IMDB dataset. For the initial part of the project, we are concerned with constructing a directed actor network and later on we shift focus to an undirected movie network.

Before we construct the networks, the problem requires us to perform some preprocessing on the data we are given. The contents of the actor and actress files are of the format ***{actor1}\t\t{movie1}\t\t{movie2}\t\t{movie3}...***. We need to separate these entries and store them into appropriate data structures to capture all the information from these files.

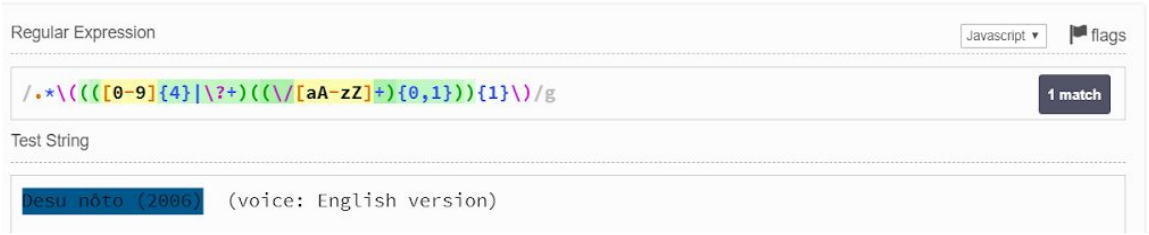
Additionally, the movie names in the actor and actress files is unclean and contains some movies such as:

Alchemy (2005/I) (uncredited)

Mui dong bin wan si (2007) (as Jolie Chan)

Desu nôto (2006) (voice: English version)

These movie names need to be cleaned so as to remove the extra information at the end, hence preserving only the movie name. To perform this cleaning, we make use of the following regular expression:



After cleaning the movie names, we have to further filter the file to remove actors and actresses which have acted in less than 10 movies.

To optimize the processing time for the preprocessing steps, we have created a mapping for movies and actors, i.e., mapping actor names to IDs and movie names

to IDs and further made used of dictionaries and named tuples for storing various combinations.

We also observed some duplicate entries for actors and actresses which we have merged in order to get proper results.

After performing the above preprocessing steps, we get the following results:

Number of Entries in Actor File	2167653
Number of Entries in Actress File	1182813
Number of Actors and Actresses	113110
Number of Unique Movies	468252

We observe that out of a combined 3 million actor and actress records, only a fraction of them are relevant to our computations. There are approximately 100k relevant actors and actresses which have 10 or more movies to their name.

Furthermore, after performing the cleaning on the movies, we obtain approximately 468k unique movies.

1.1. Directed actor/actress network creation

We will use the processed text file to create the directed actor/actress network. The nodes of the network are the actor/actress and there are weighted edges between the nodes in the network. The weights of the edges are given by equation 1

$$w_{i \to j} = \frac{|S_i \cap S_j|}{|S_i|}$$

where S_i is the set of movies in which actor/actress vi has acted in and S_j is the set of movies in which actor/actress v_j has acted in.

2) Create a weighted directed actor/actress network using the 2 processed text file and equation 1. Plot the in-degree distribution of the actor/actress network. Briefly comment on the in-degree distribution.

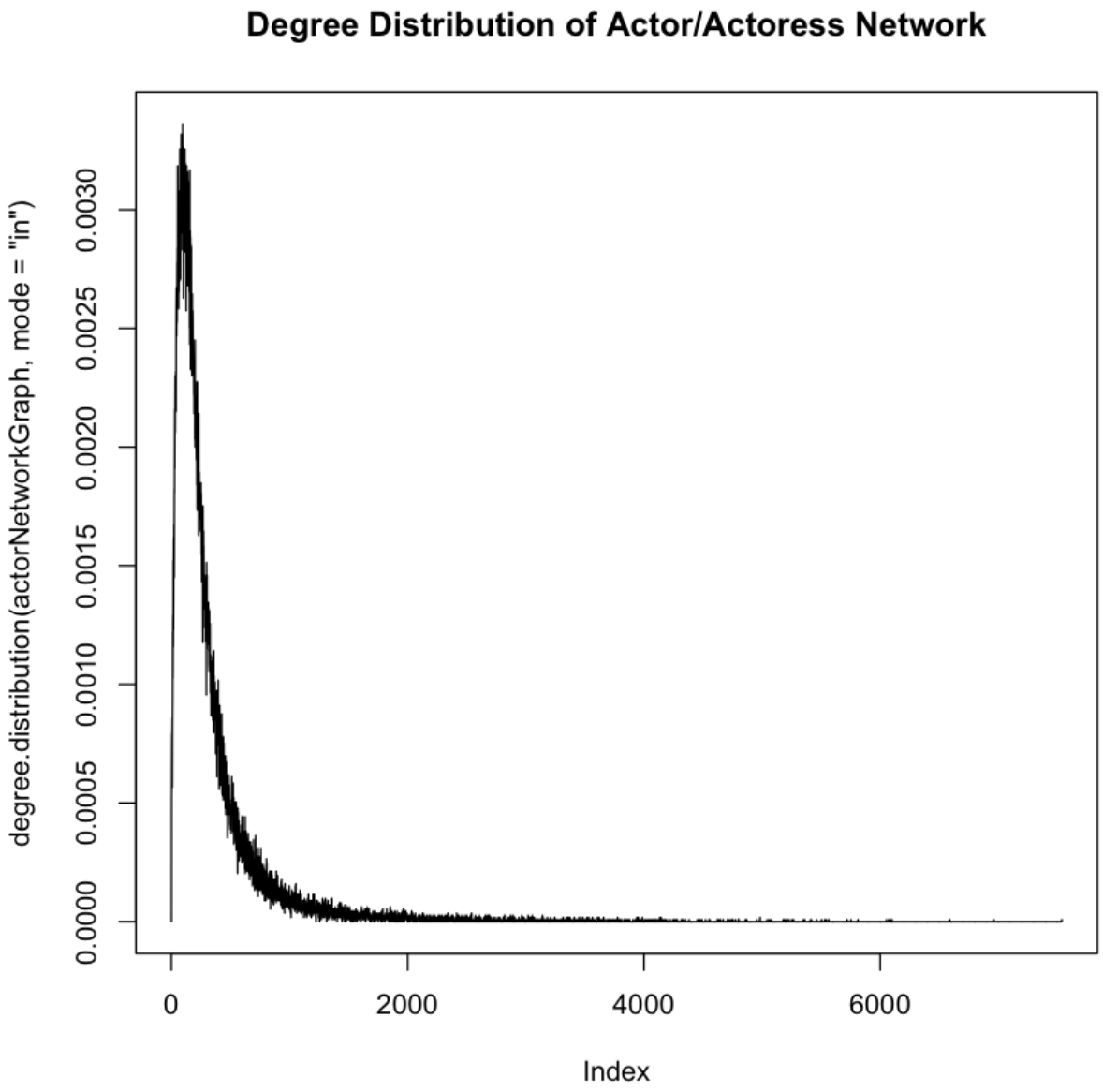
Ans:

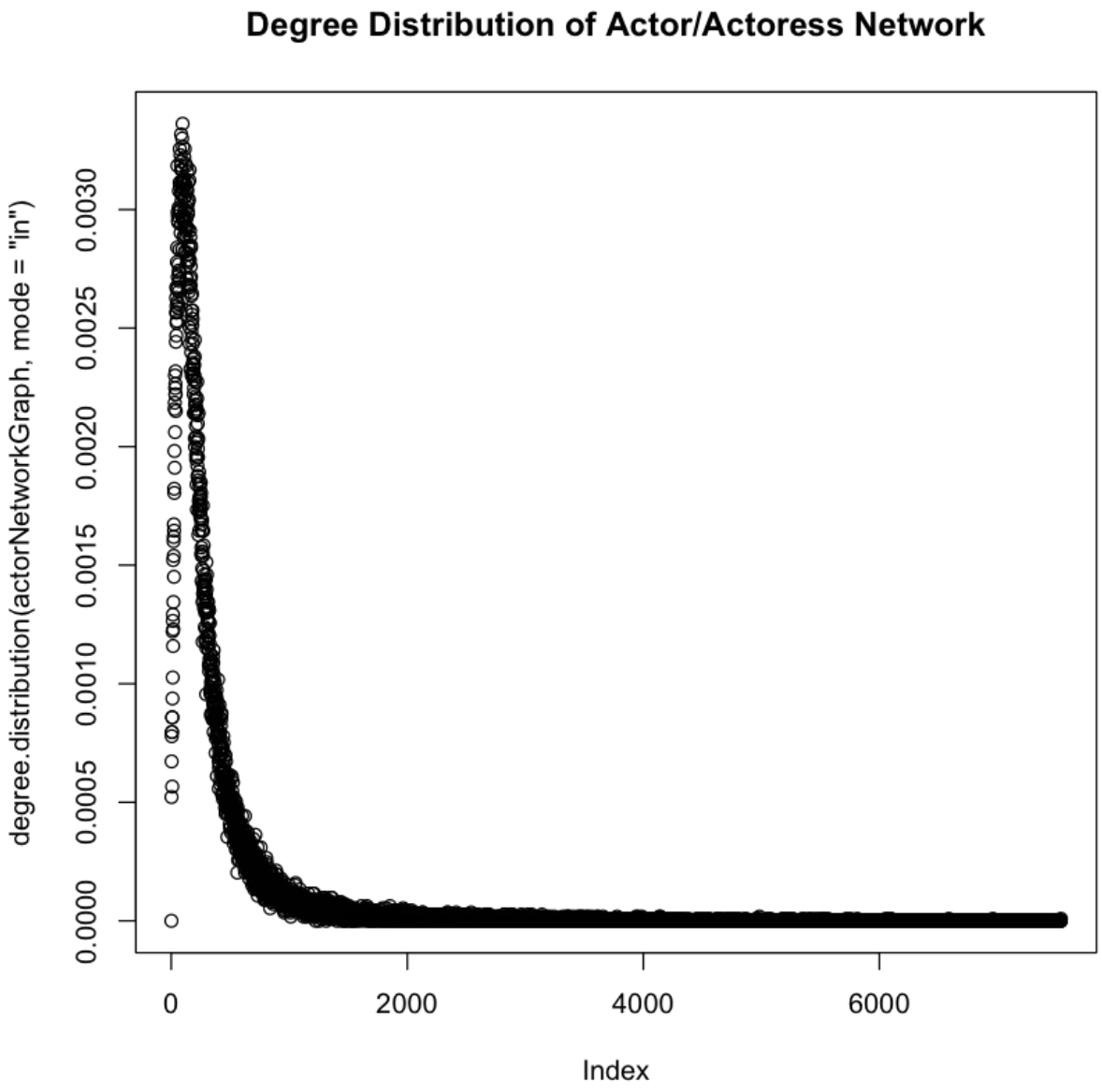
After finding out the relevant actors and actresses and their movie information from the previous part, we create an edge list for the network of the following format:

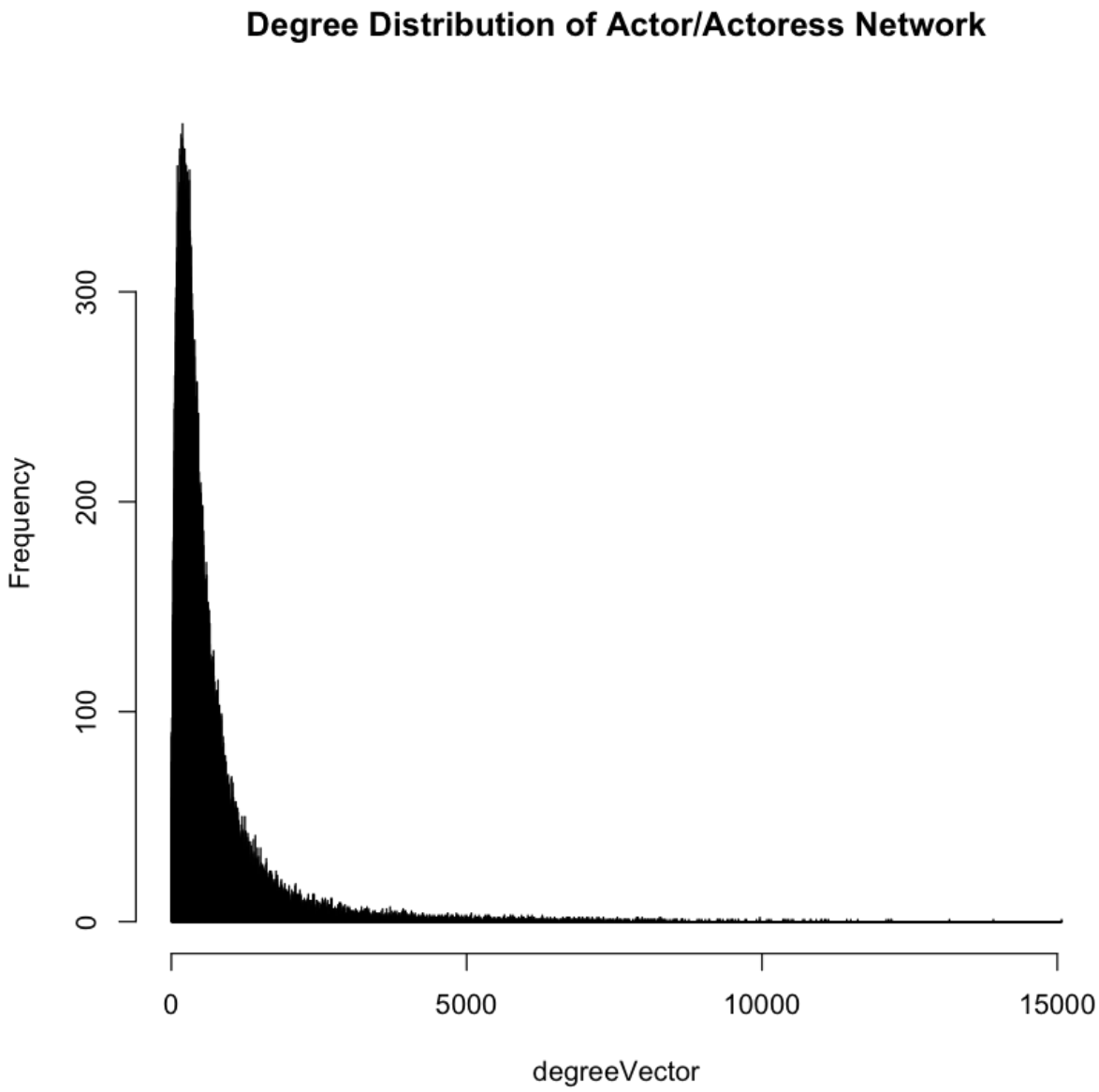
Actor1	**Actor2**	**Weight**
1	3300	0.5
3300	1	0.67
1	55	0.21

On creating the edge list and the corresponding network, we obtain a total of ***35 million edges (35467540)***.

The normalized degree distribution (line and scatter plots) and the histogram of the degree vector are shown below:


Degree Distribution of Actor/Actoress Network
degree.distribution(actorNetworkGraph, mode = "in")
0.0000
0.0005
0.0010
0.0015
0.0020
0.0025
0.0030
0
2000
4000
6000
Index


Degree Distribution of Actor/Actoress Network
degree.distribution(actorNetworkGraph, mode = "in")
0.0000
0.0005
0.0010
0.0015
0.0020
0.0025
0.0030
0
2000
4000
6000
Index



On observing the degree distribution, we observe that there are a large number of actors/actresses having a smaller degree and as the degree increases, the number of actors/actresses associated with them decreases.

The large volume of actors/actresses having smaller degree indicates that for most of the performers, the intersection between their movies is not significant, i.e., actors don't act in movies with every other actor and usually act with a few actors only in many movies.

1.2. Actor pairings

3) **Design a simple algorithm to find the actor pairings. To be specific, your algorithm should take as input one of the actors listed above and should return the name of the actor with whom the input actor prefers to work the most. Run your algorithm for the actors listed above and report the actor names returned by your algorithm. Also for each pair, report the (input actor, output actor) edge weight. Does all the actor pairing make sense?**

Ans:

In this part, we have used the actor network graph that we have created in the previous part. More specifically, we have used a simple algorithm based on the immediate neighbors of an actor. The actor network is generated based on the common movies between 2 actors and the edge weights reflect this. So to find the most preferred actor we find the immediate neighbors of an actor and the most preferred actor is the one with the highest edge weight among them. We get the below results.

Input Actor	Output Preferred Actor	Edge Weight
Cruise, Tom	Kidman, Nicole	0.1746032
Watson, Emma (II)	Radcliffe, Daniel	0.52
Clooney, George	Damon, Matt	0.119403
Hanks, Tom	Allen, Tim (I)	0.1012658
Johnson, Dwayne (I)	Austin, Steve (IV) Calaway, Mark Levesque, Paul (I)	0.2051282
Depp, Johnny	Bonham Carter, Helena	0.08163265
Smith, Will (I)	Foster, Darrell	0.122449
Streep, Meryl	De Niro, Robert Kline, Kevin (I)	0.06185567
DiCaprio, Leonardo	Scorsese, Martin	0.1020408
Pitt, Brad	Clooney, George	0.09859155

We see that the actor pairing using this method makes sense when verified with news and information from the internet.

Emma Watson and Daniel Radcliffe have worked in the Harry Potter series.
Tom Hanks and Tim Allen are the voice actors for the Toy Story series of animated movies.
Johnny Depp has worked with Helena Carter in a lot of movies.
The other pairs also have similar such relationships.

1.3. Actor rankings

In this section, we will extract the top 10 actor/actress from the network.

4) Use the google's pagerank algorithm to find the top 10 actor/actress in the network. Report the top 10 actor/actress and also the number of movies and the in-degree of each of the actor/actress in the top 10 list. Does the top 10 list have any actor/actress listed in the previous section? If it does not have any of the actor/actress listed in the previous section, please provide an explanation for this phenomenon.

Ans:

We run the pagerank algorithm on the actor graph network. The results for this are presented below.

Actor/Actress	PageRank score	In-degree	Number of Movies
Flowers, Bess	0.00023518	7537	828
Tatasciore, Fred	0.00019888	3951	353
Harris, Sam (II)	0.00019725	6960	600
Blum, Steve (IX)	0.00019543	3313	373
Miller, Harold (I)	0.00017275	6587	561
Jeremy, Ron	0.0001586	2905	637
Phelps, Lee (I)	0.00015735	5563	647
Lowenthal, Yuri	0.00015657	2656	317

Downes, Robin Atkin	0.00015171	2951	267
O'Connor, Frank (I)	0.00014698	5502	623

From the above list, we see that the actors with top 10 page rank values are different from those in the popular list. Based on how the graph is constructed, we can easily see why this is so. The actor graph is generated based on the relation between the actors and in how many movies they have worked together. We did some analysis of the top ones and found that Bess Flowers was an American actress best known for her work as an extra in hundreds of films while Tatasciore, Fred is a voice actor. Because of this these actors have worked in a lot of movies as well as with a lot of actors and hence have a very high in-degree. The popular 10 actors have in comparison worked in fewer movies and have lower in-degree.

The PageRank works by counting the number and quality of links to a vertex to determine a rough estimate of how important the vertex is. The underlying assumption is that more important vertices are likely to receive more links from other vertices. With this as the core working of the algorithm it is clear that the algorithm will consider the quantity and not exactly the quality or popularity of the actors and hence will assign high pagerank scores to the vertices which have a high number of connections, especially a high in-degree. Hence we get a different list for top pagerank score actors as compared to the popular actors.

5) Report the pagerank scores of the actor/actress listed in the previous section. Also, report the number of movies each of these actor/actress have acted in and also their in-degree.

Ans:

As mentioned in the previous part, these popular actors have worked in fewer movies and have a much lower in-degree. Hence they have a lower pagerank score. The results for each of them are compiled in the table below.

Actor/Actress	PageRank score	In-degree	Number of Movies
Cruise, Tom	3.98E-05	1651	63
Watson, Emma (II)	1.75E-05	453	`25

Clooney, George	4.00E-05	1573	67
Hanks, Tom	5.10E-05	2063	79
Johnson, Dwayne (I)	4.19E-05	1355	78
Depp, Johnny	5.38E-05	2144	98
Smith, Will (I)	3.20E-05	1318	49
Streep, Meryl	3.96E-05	1594	97
DiCaprio, Leonardo	3.17E-05	1300	49
Pitt, Brad	4.30E-05	1740	71

2. Movie network

In this part, we will create an undirected movie network and then explore the various structural properties of the network.

2.1. Undirected movie network creation

We will use the processed text files from the previous section to create the movie network. The nodes of the network are the movies and there are weighted edges between the nodes in the network. To reduce the size of the network, we will only consider movies that has at least 5 actor/actress in it. The weights of the edges are given by below equation.

$$w_{i \to j} = \frac{|A_i \cap A_j|}{|A_i \cup A_j|}$$

where A_i is the set of actors in movie v_i and A_j is the set of actors in movie v_j . Since,

$$w_{i \to j} = w_{j \to i}$$

so we have an undirected network.

6) **Create a weighted undirected movie network using equation 2. Plot the degree distribution of the movie network. Briefly comment on the degree distribution.**

Ans:

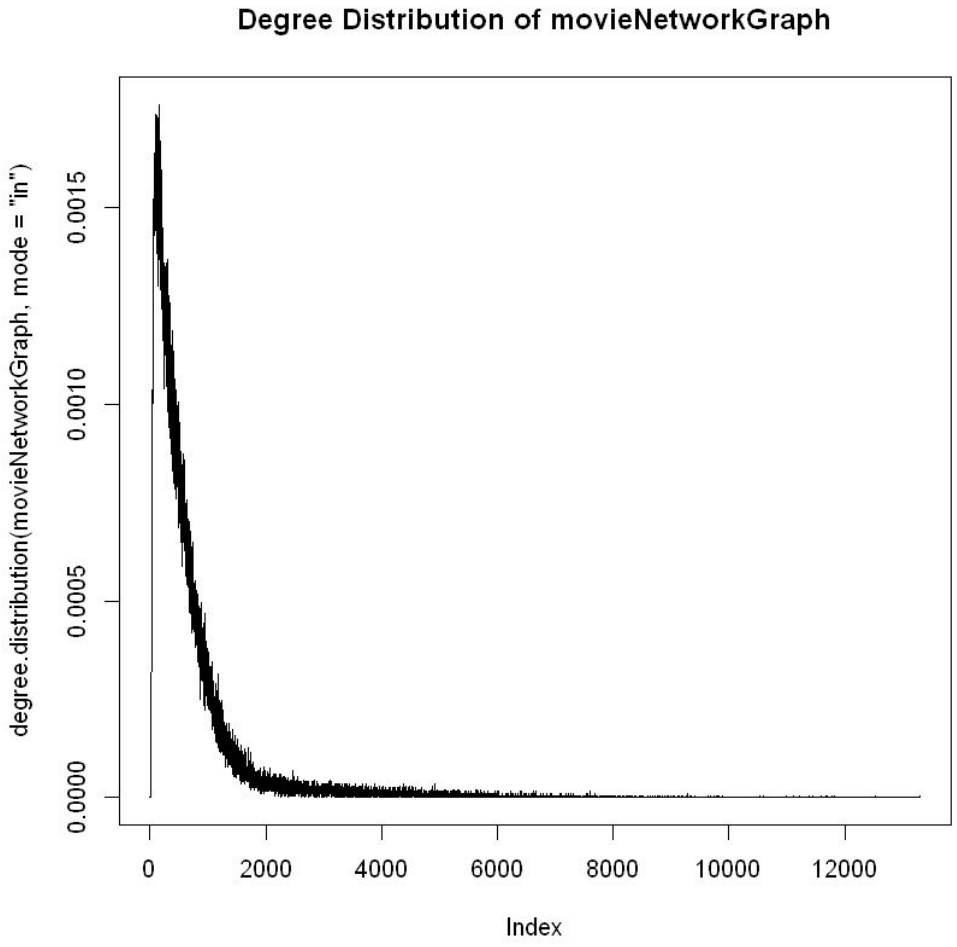
In this and the following parts, we create an undirected movie network based on actor information available for them. Before we create the edge list, we are required to filter out movies which have less than 5 actors associated with it. After this processing, we get around ***200k movies (203587)***.

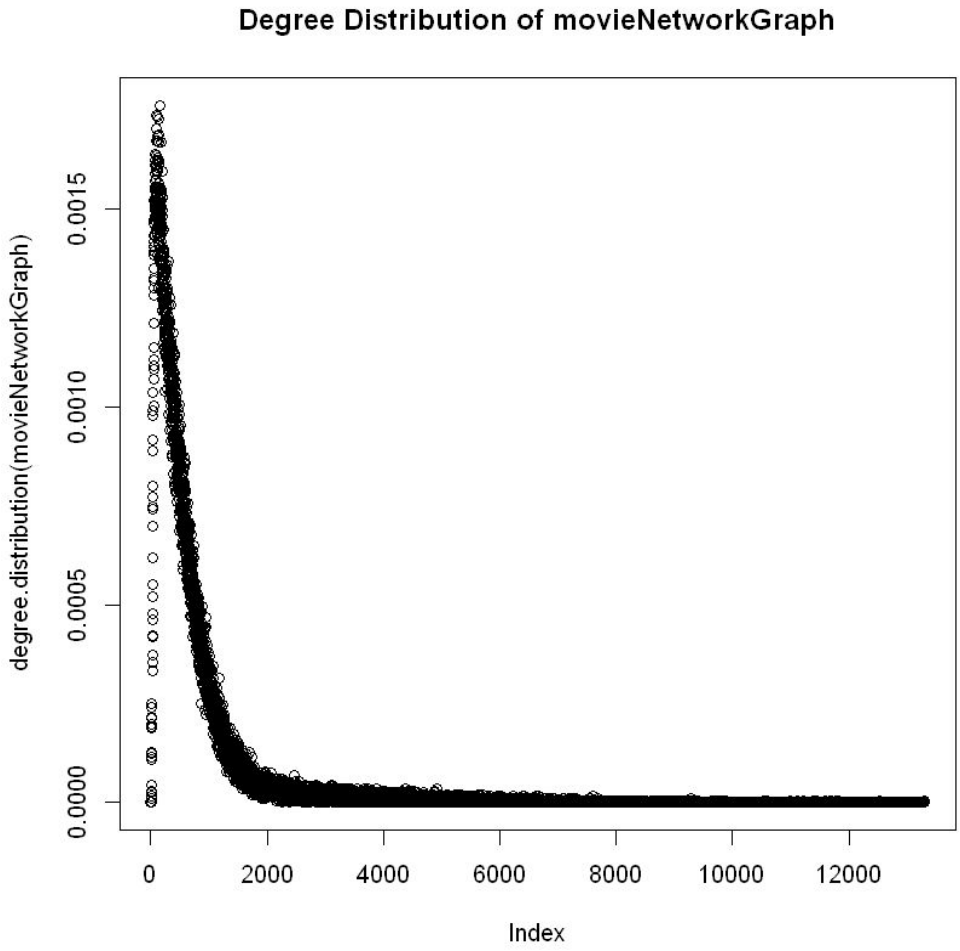
After performing this processing and using the actor and movie information achieved in earlier parts,, we create an edge list for the network of the following format:

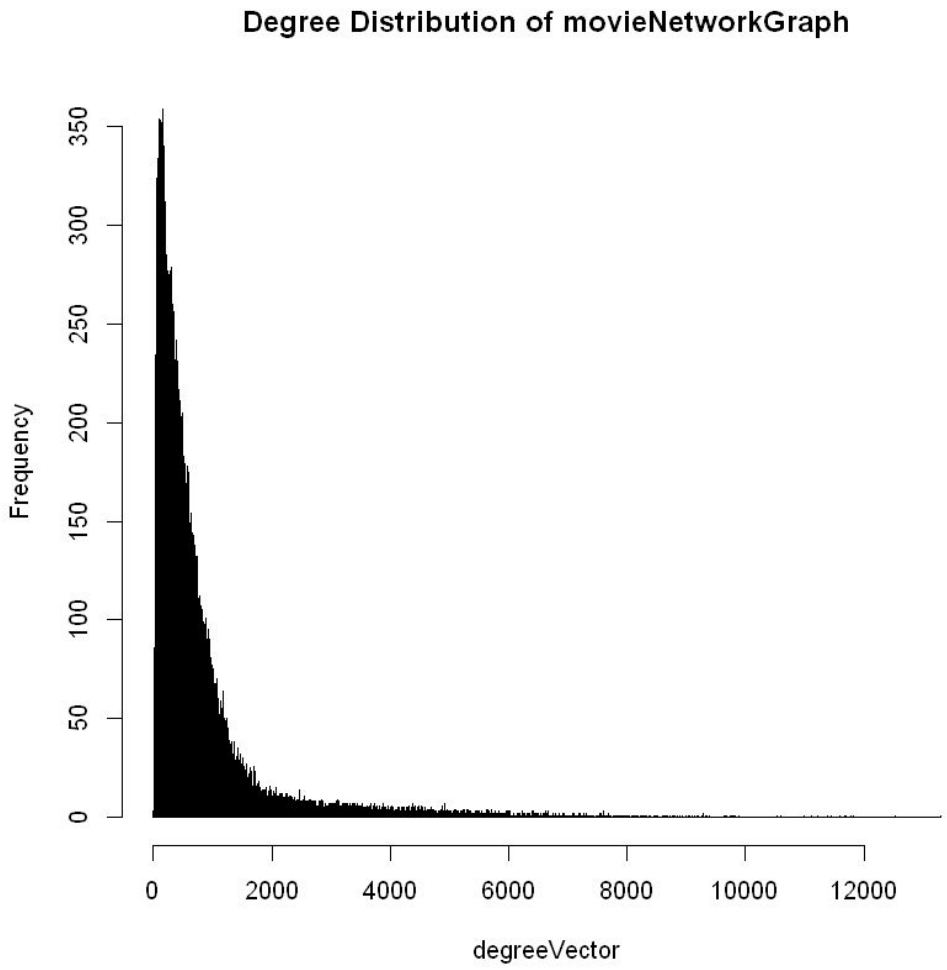
Movie1	**Movie2**	**Weight**
1	65	0.5
1	799	0.1299
1	5000	0.7899

On creating the edge list and the corresponding network, we obtain a total of ***66 million edges (66527900)***.

The normalized degree distribution (line and scatter plots) and the histogram of the degree vector are shown below:

Line Plot of Normalized Degree (As it is an undirected graph, in degree is equivalent to out degree and the overall degree of the network)	

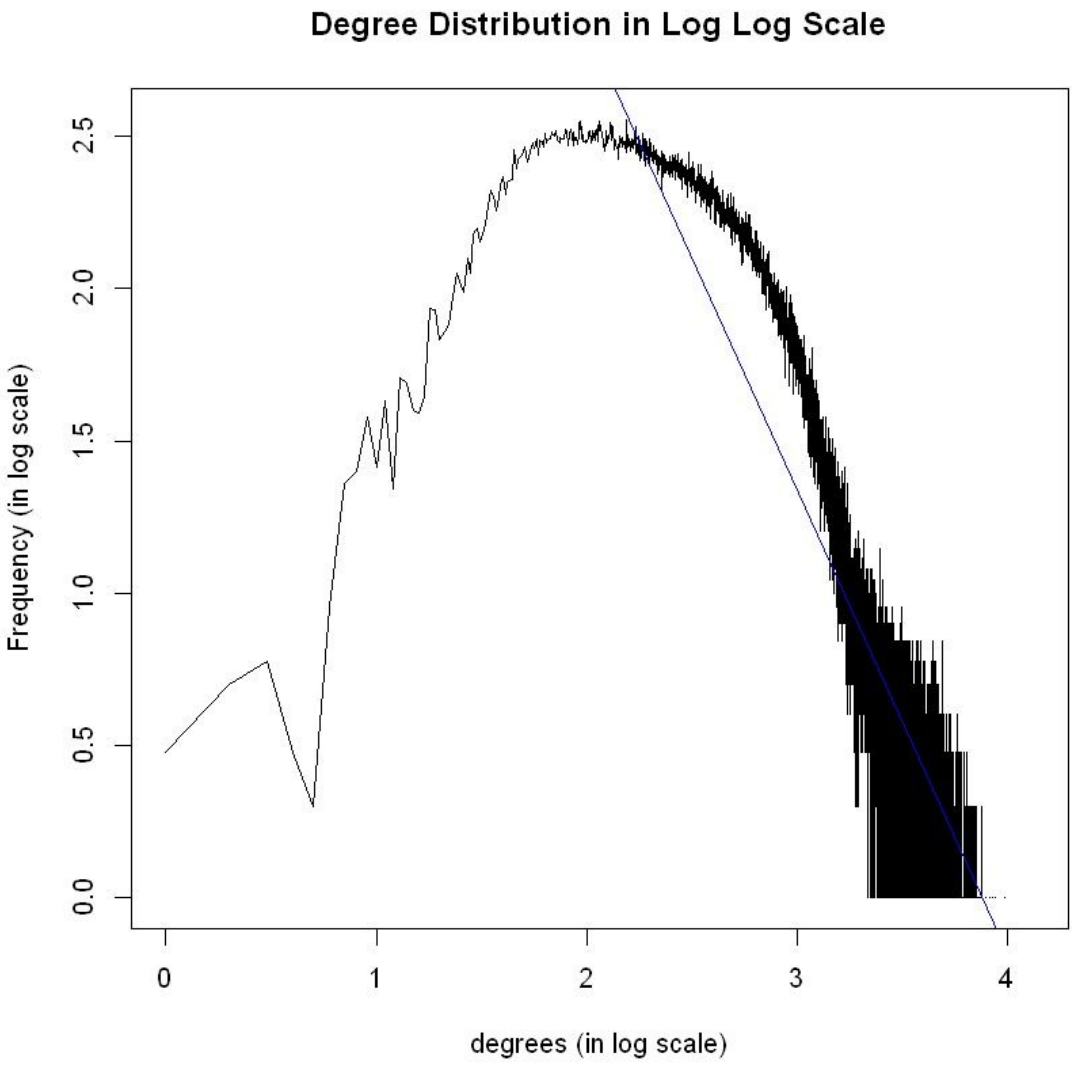
Scatter Plot of Normalized Degree	

Histogram of the Degree Vector	

Similar to the actor network created earlier, we observe that there are a large number of movies having a smaller degree and as the degree increases, the number of movies associated with them decreases.

The large volume of movies having smaller degree indicates that there are not any common actors between a majority of movies which prevents the formation of edges between a lot of the movies.

On looking at the degree distribution, we realized that it mirrored the distribution of some power law networks encountered earlier in the course and we wanted to see if our movie network also follows such a distribution. So we plotted the degree distribution in the log log scale and tried fitting a line through it. We got the following plot:



On looking at the log-log plot, it is not a straight line fit and hence not a power law fit.

2.2. Communities in Movie Network

In this part, we will extract the communities in the movie network and explore their relationship with the movie genre. For this part you will need to load the movie_genre.txt file.

7) Use the Fast Greedy community detection algorithm to find the communities in the movie network. Pick 10 communities and for each community plot the distribution of the genres of the movies in the community.

Ans:

Fast greedy is a hierarchical agglomeration algorithm for detecting community structure which is faster than many competing algorithms. It tries to optimize the modularity in a greedy manner. Its running time on a network with 'n' vertices and 'm' edges is O(m*d*log n) where 'd' is the depth of the dendrogram describing the community structure.

$$e_{ij} = \frac{1}{2m} \sum_{vw} A_{vw} \delta(c_v, i) \delta(c_w, j)$$

The above represent the fraction of edges that connect vertices in community i to vertices in community j and

$$a_i = \frac{1}{2m} \sum_{v} k_v \delta(c_v, i)$$

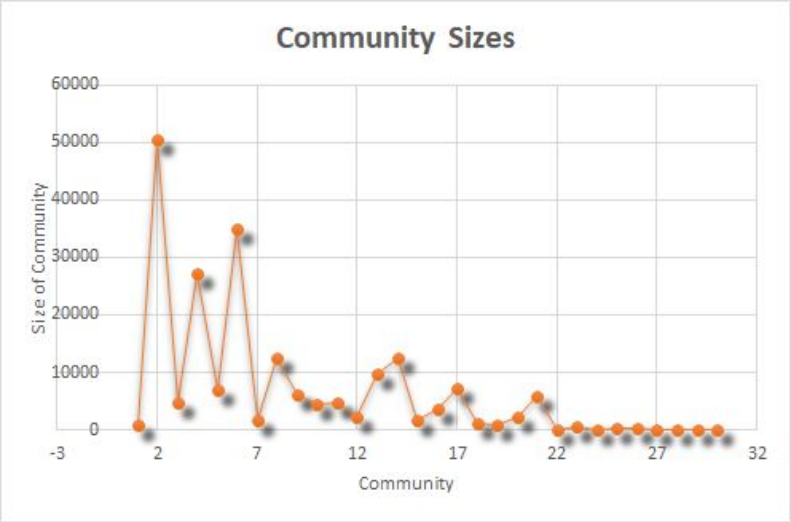
be the fraction of edges that are attached to vertices in community i, then the operation of the algorithm involves finding the changes in

$$Q = \frac{1}{2m} \sum_{vw} \left[A_{vw} - \frac{k_v k_w}{2m} \right] \sum_{i} \delta(c_v, i) \delta(c_w, i)$$

that would result from the amalgamation of each pair of communities, choosing the largest of them, and performing the corresponding amalgamation.

Fast greedy algorithm tends to output lower number of communities due to its hierarchical agglomeration style of operation. It also tends to output larger size of communities and almost no communities which have very few nodes. This leads to a relatively higher modularity score for this algorithm for all node IDs.

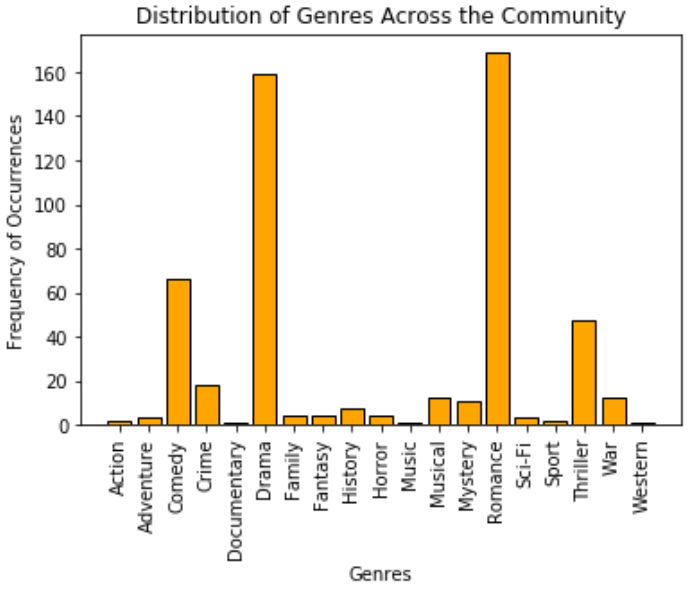
On performing Fastgreedy community detection on our movie network graph, we obtain the following results:

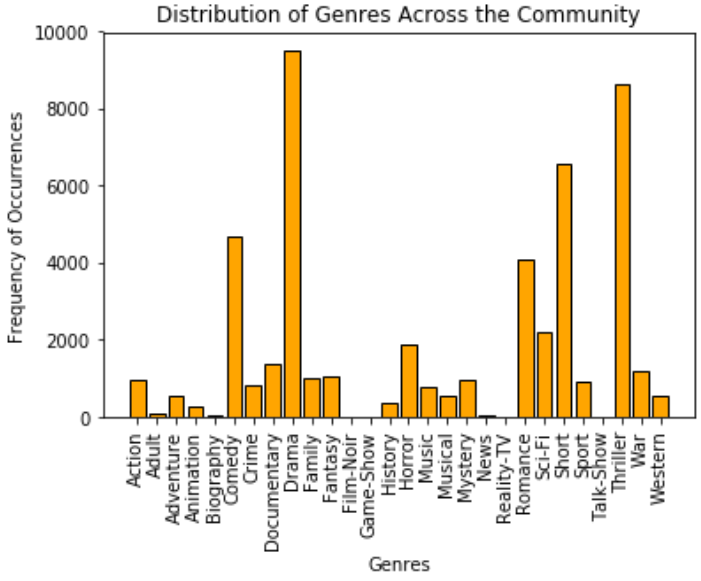
Number of Communities	30
Sizes of communities	764 50475 4821 27211 6928 34936 1595 12539 6196 4466 4876 2269 9628 12636 1792 3516 7272 1149 844 2115 5938 23 623 14 433 405 17 18 72 14
	
Modularity	0.79934022007948

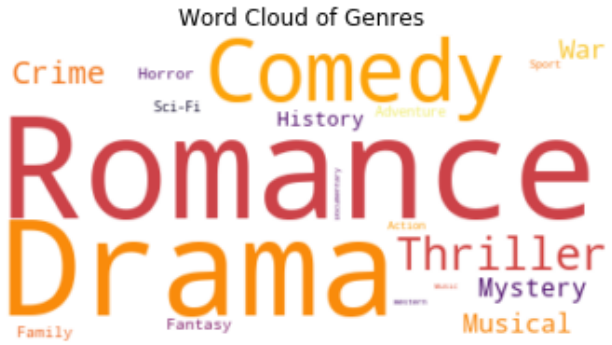
We get 30 communities of varying sizes. The largest community contains 50k movies and the smallest community has 14 movies. We also observe that there are very few communities which have large sizes and the majority of them are smaller in size. We also obtain a high value of modularity indicating the presence of dense intra community connections and sparse inter community connections.

We further analyse genre information obtained from movie_genre.txt and plot the distribution of the genres present in each community. The results of our analysis are presented below:

Community # 1 *Size of Community: 764* *Movies Having Genre Information: 526*	**Community # 2** *Size of Community: 50475* *Movies Having Genre Information: 49143*

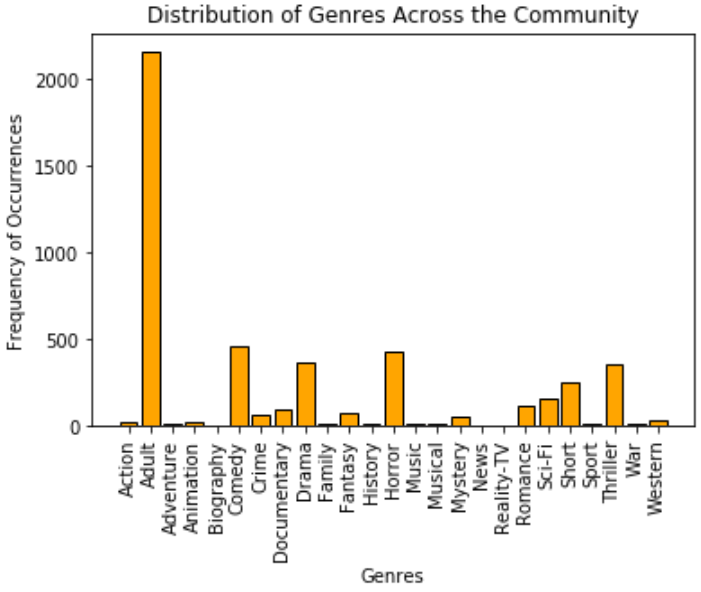


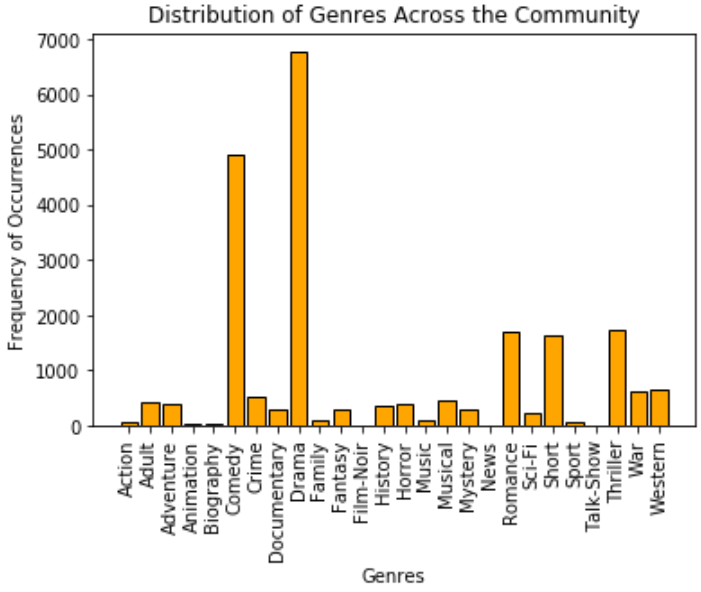


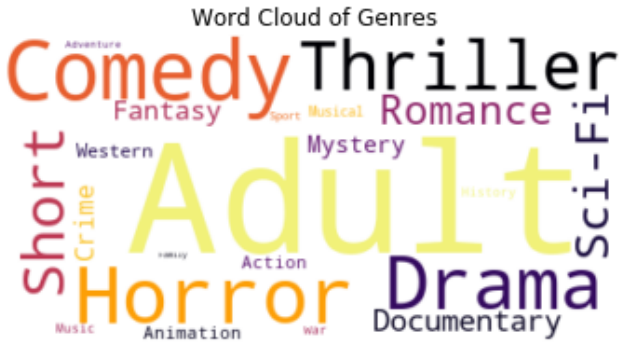


Community # 3
Size of Community: 4821
Movies Having Genre Information: 4714

Community # 4
Size of Community: 27211
Movies Having Genre Information: 22171

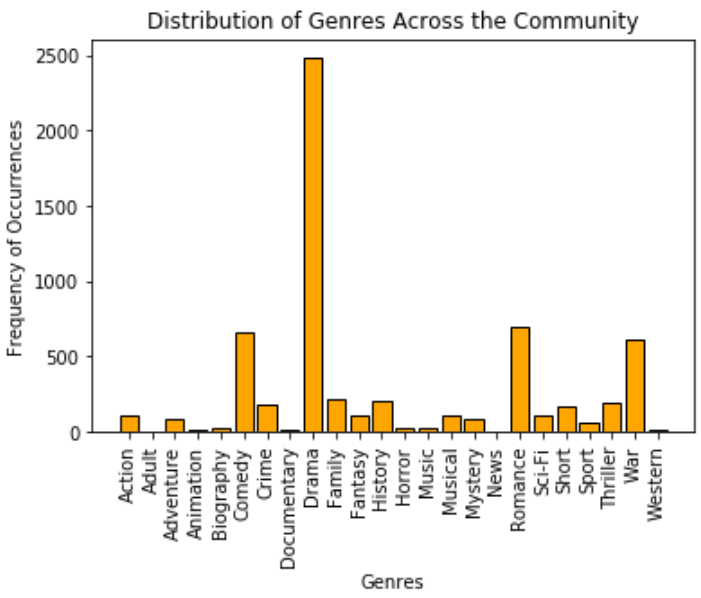


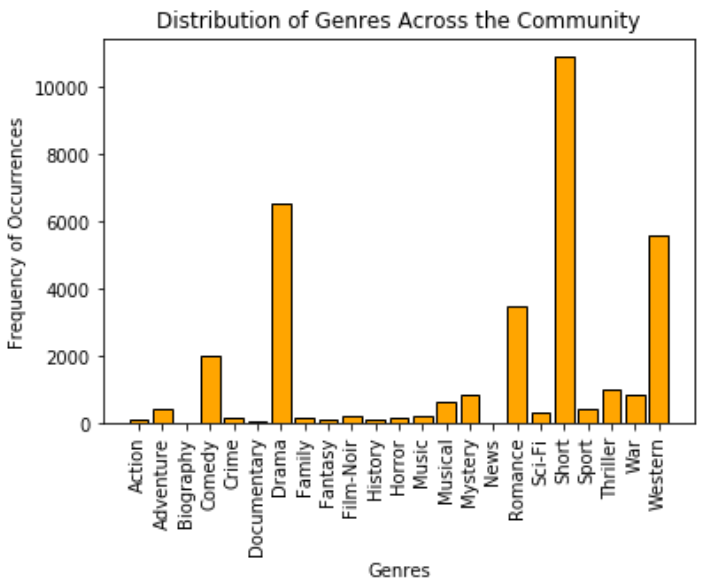




Community # 5
Size of Community: 6928
Movies Having Genre Information: 6204

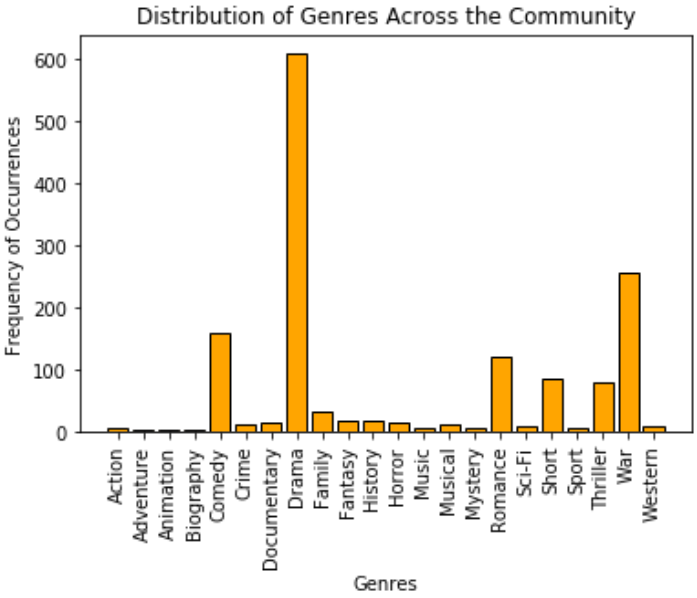
Community # 6
Size of Community: 34936
Movies Having Genre Information: 34275

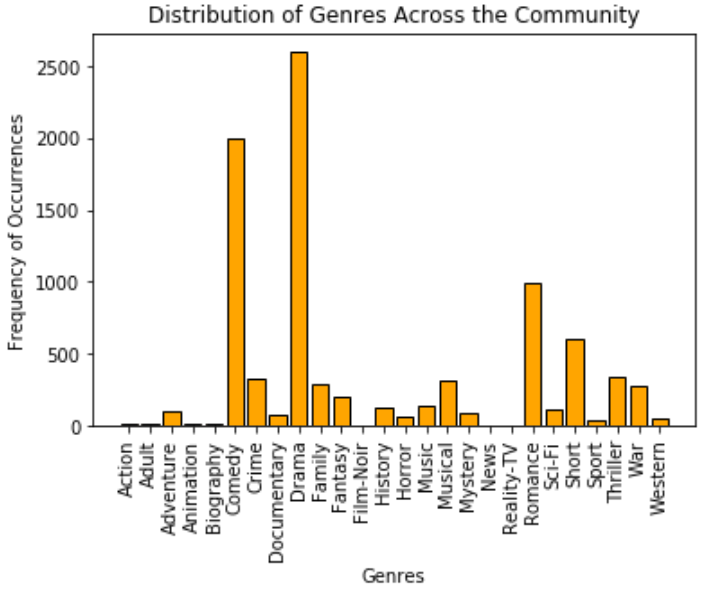


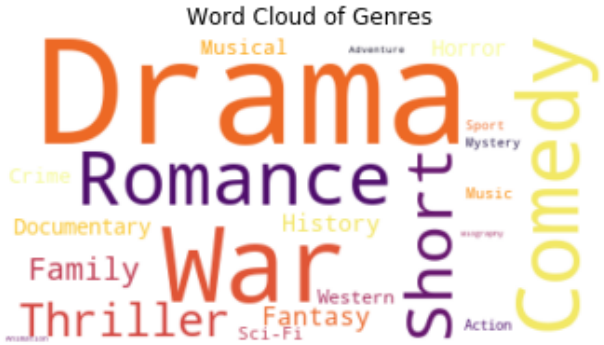


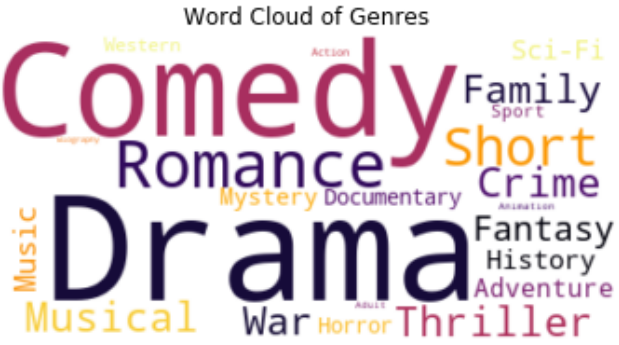
Community # 7
Size of Community: 1595
Movies Having Genre Information: 1479

Community # 8
Size of Community: 12539
Movies Having Genre Information: 8780



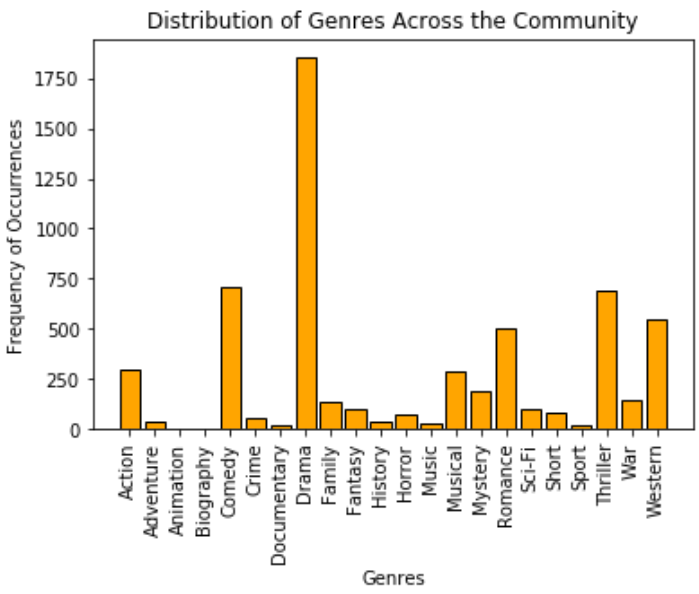


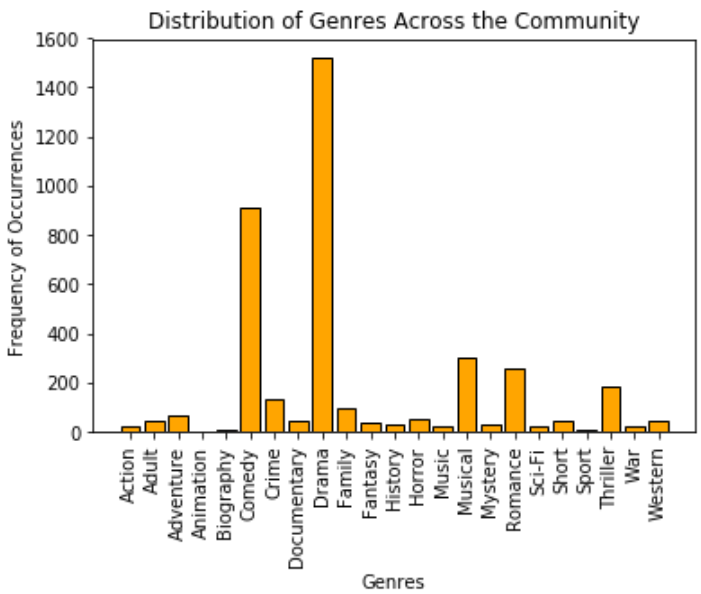


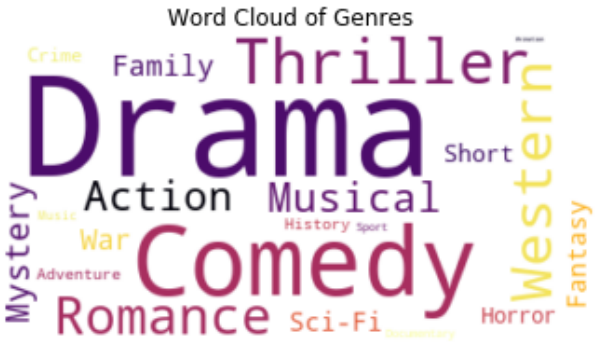


Community # 9
Size of Community: 6196
Movies Having Genre Information: 5901

Community # 10
Size of Community: 4466
Movies Having Genre Information: 3910

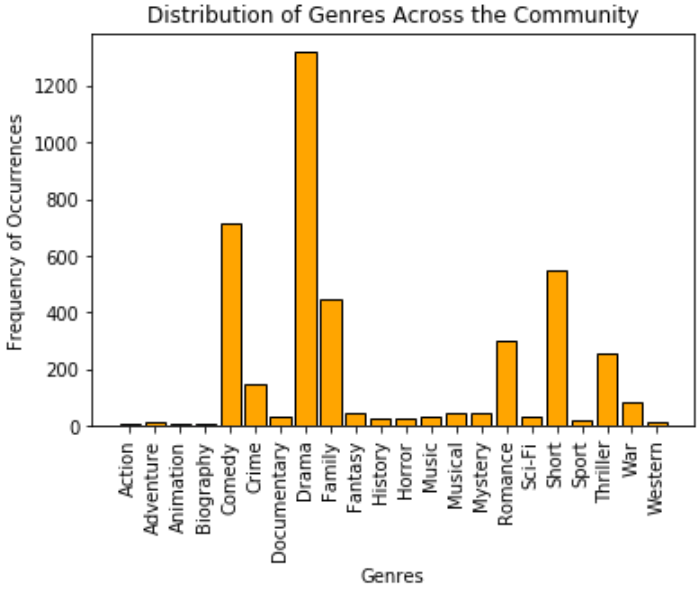


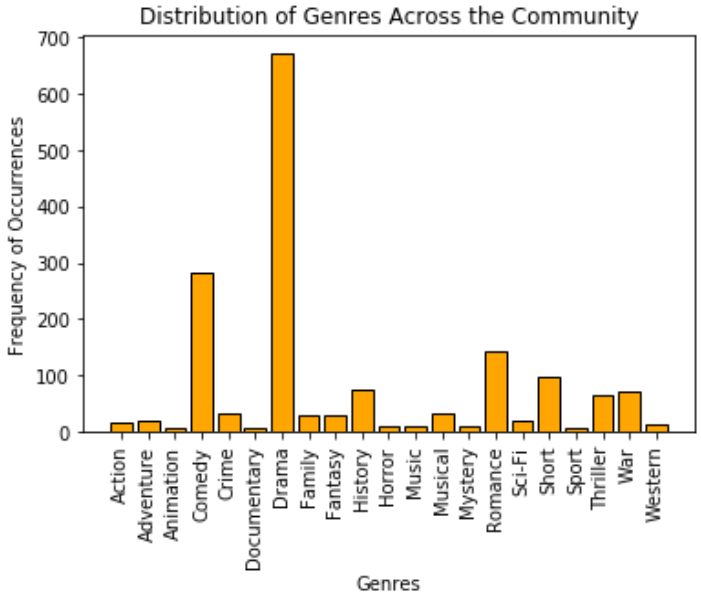


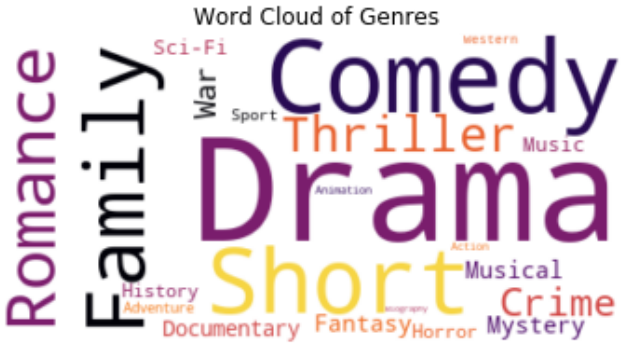


Community # 11
Size of Community: 4876
Movies Having Genre Information: 4153

Community # 12
Size of Community: 2269
Movies Having Genre Information: 1653

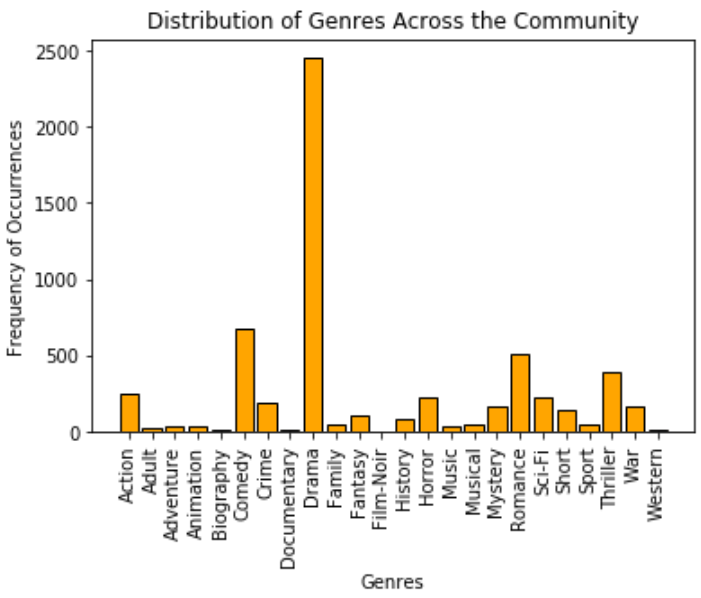


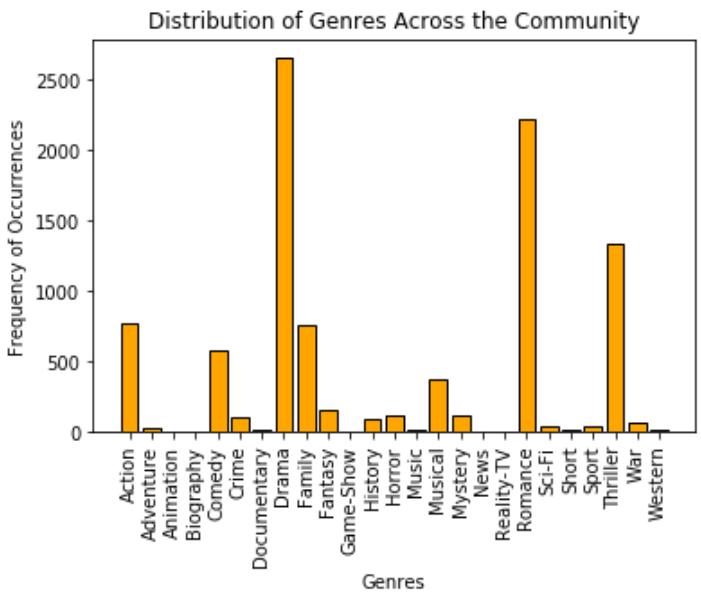




Community # 13
Size of Community: 9628
Movies Having Genre Information: 5888

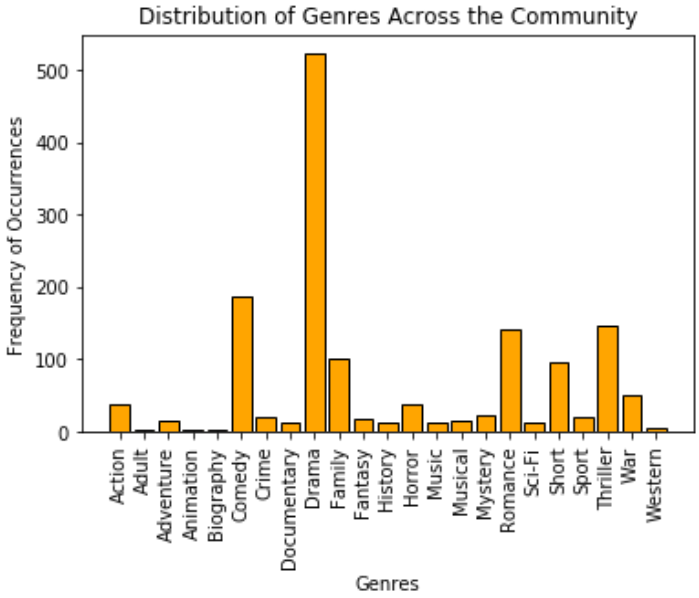
Community # 14
Size of Community: 12636
Movies Having Genre Information: 9530

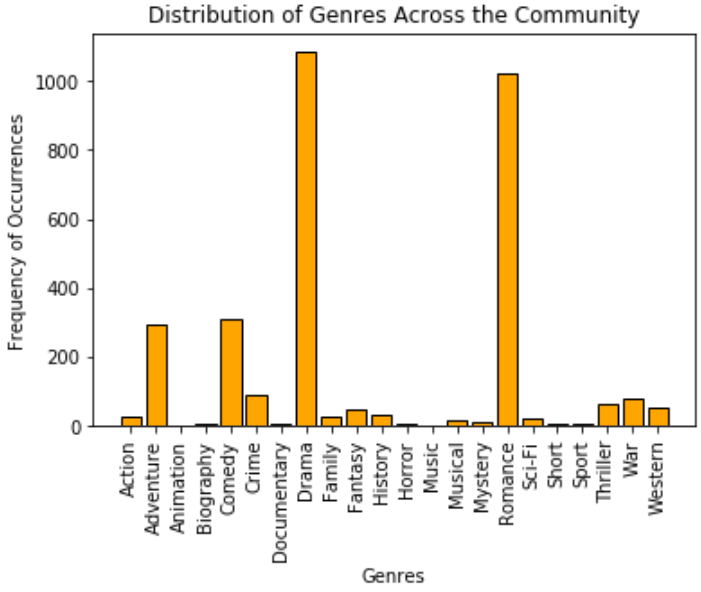


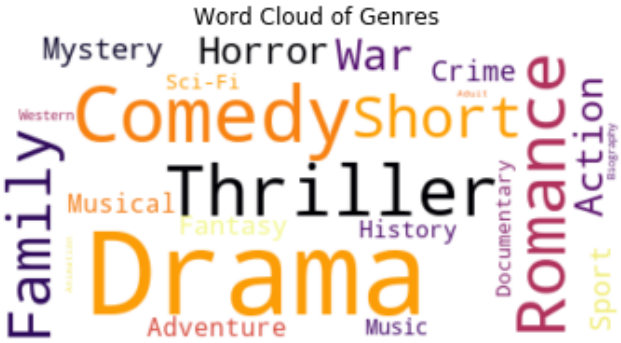


Community # 15
Size of Community: 1792
Movies Having Genre Information: 1495

Community # 16
Size of Community: 3516
Movies Having Genre Information: 3196

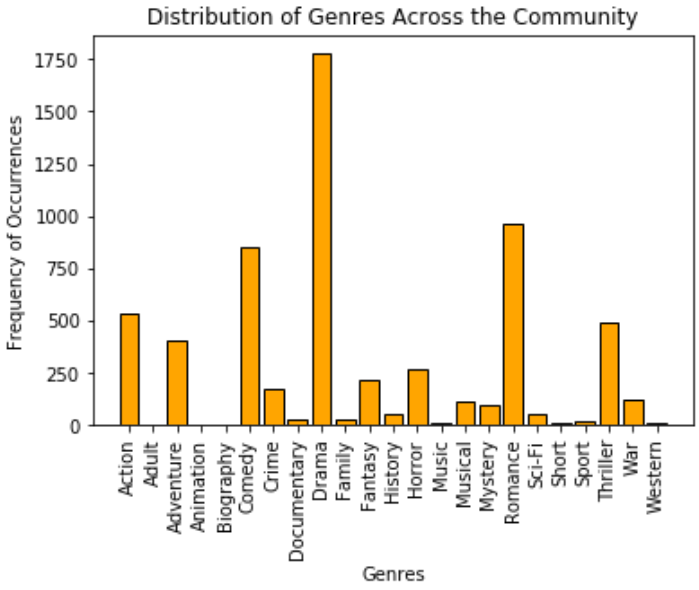


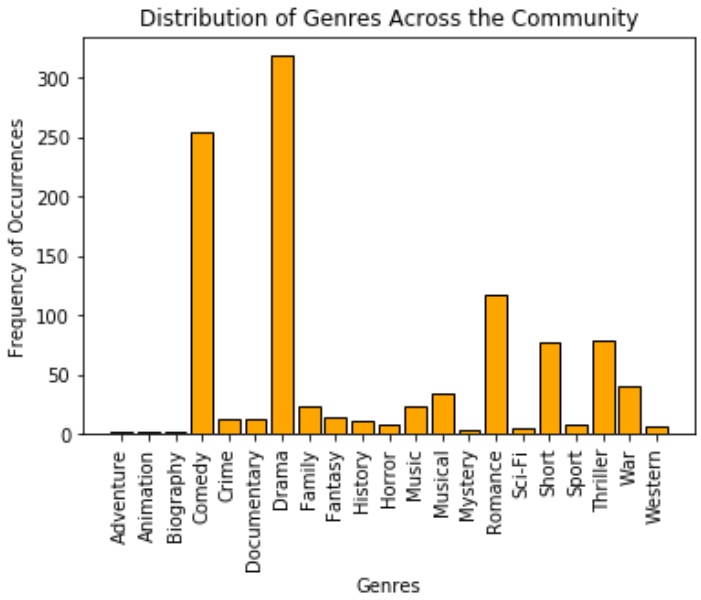


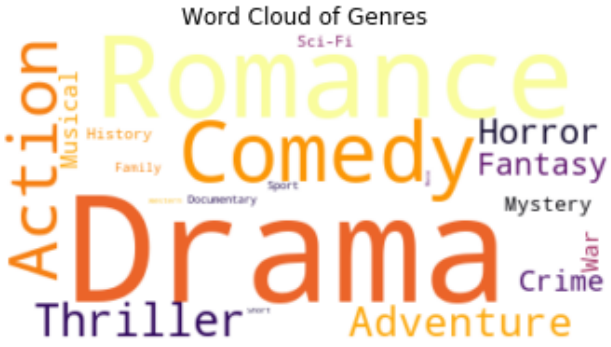


Community # 17
Size of Community: 7272
Movies Having Genre Information: 6202

Community # 18
Size of Community: 1149
Movies Having Genre Information: 1050

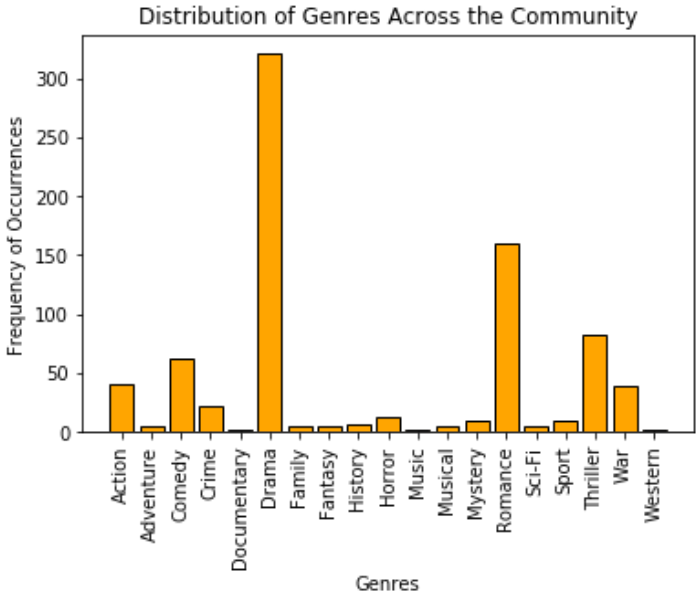


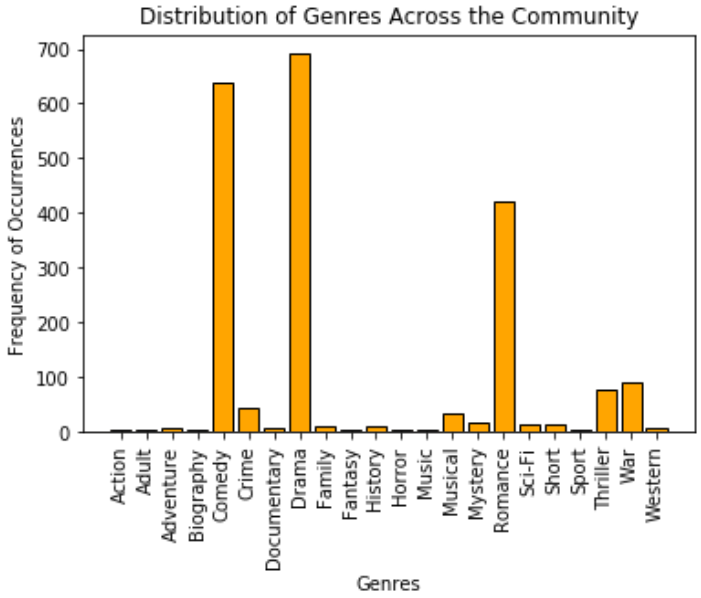




Community # 19
Size of Community: 844
Movies Having Genre Information: 792

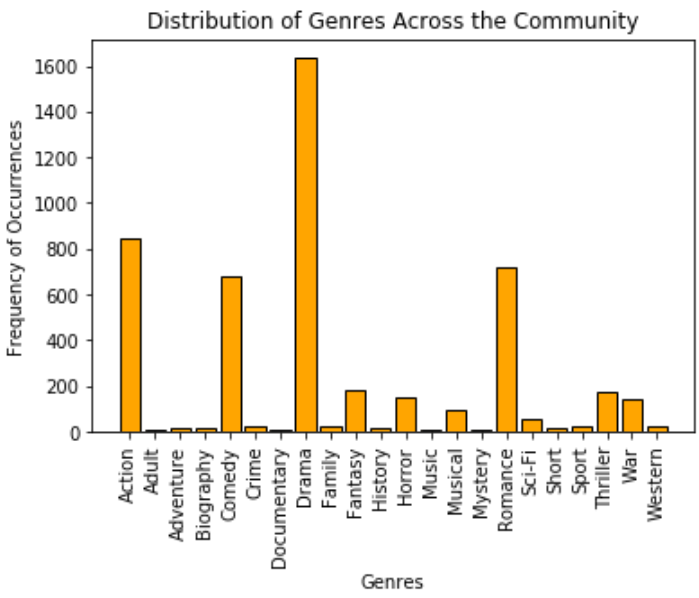
Community # 20
Size of Community: 2115
Movies Having Genre Information: 2103

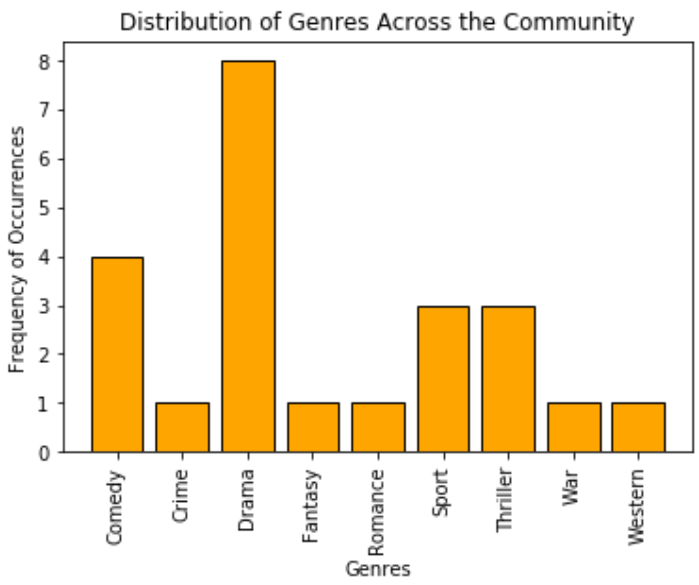


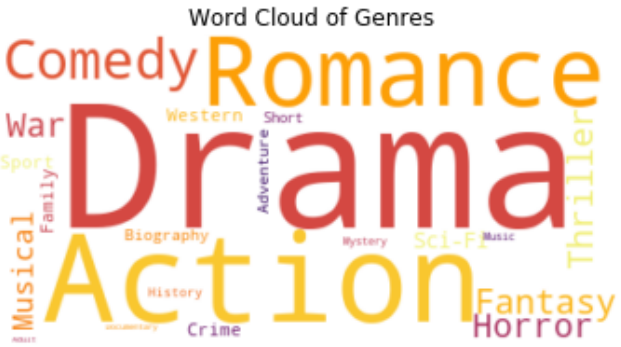


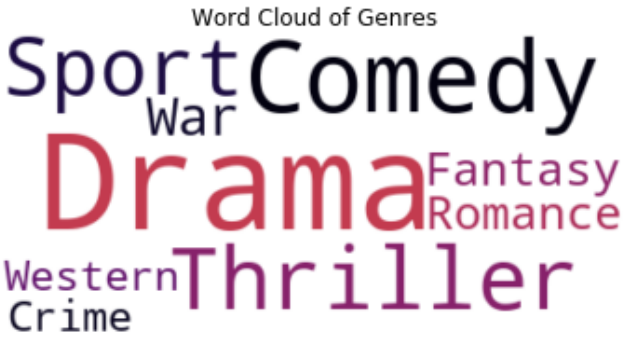
Community # 21
Size of Community: 5938
Movies Having Genre Information: 4881

Community # 22
Size of Community: 23
Movies Having Genre Information: 23



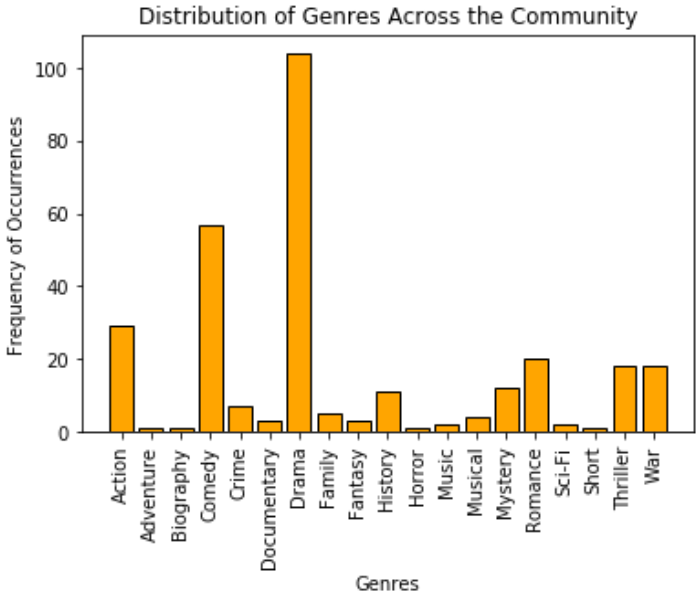


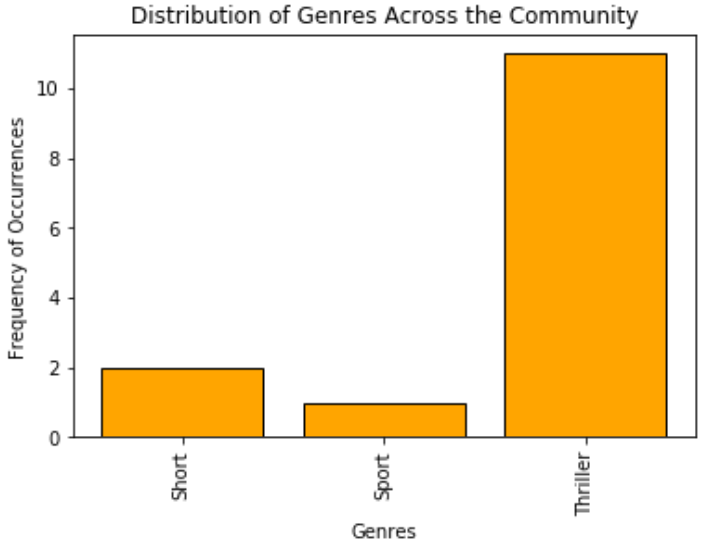




Community # 23
Size of Community: 623
Movies Having Genre Information: 299

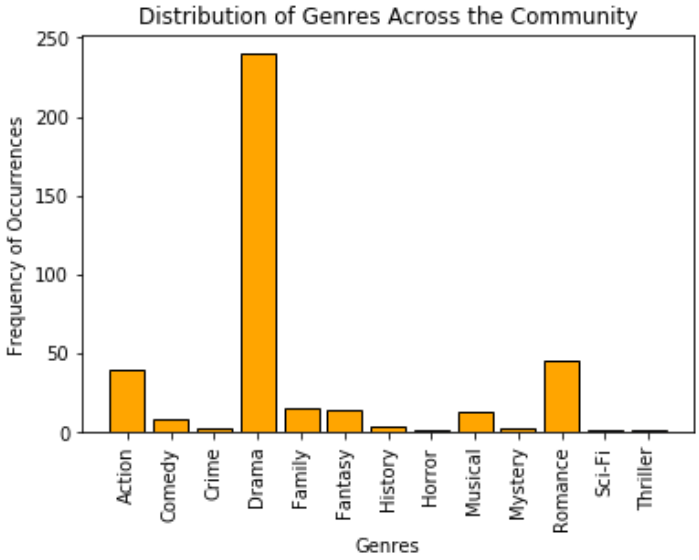
Community # 24
Size of Community: 14
Movies Having Genre Information: 14

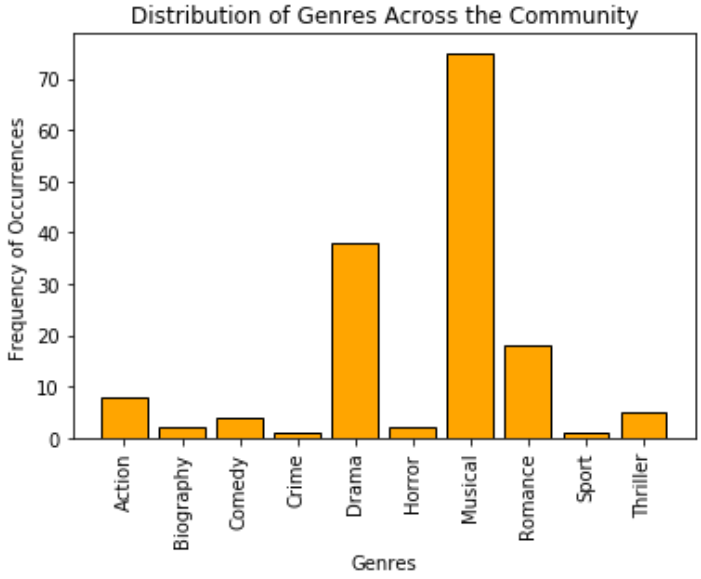


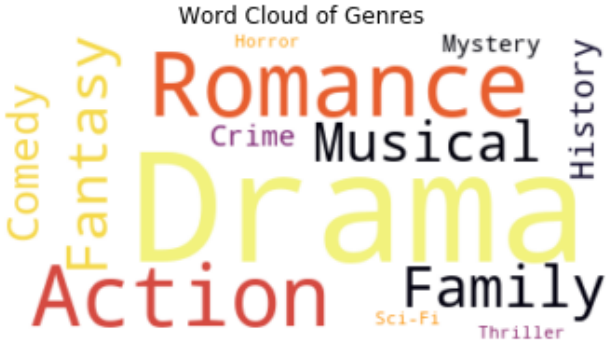


Community # 25
Size of Community: 433
Movies Having Genre Information: 387

Community # 26
Size of Community: 405
Movies Having Genre Information: 154

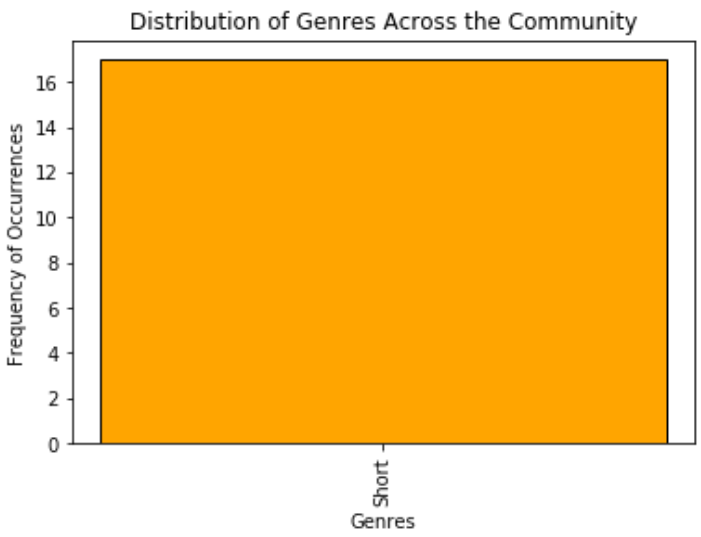
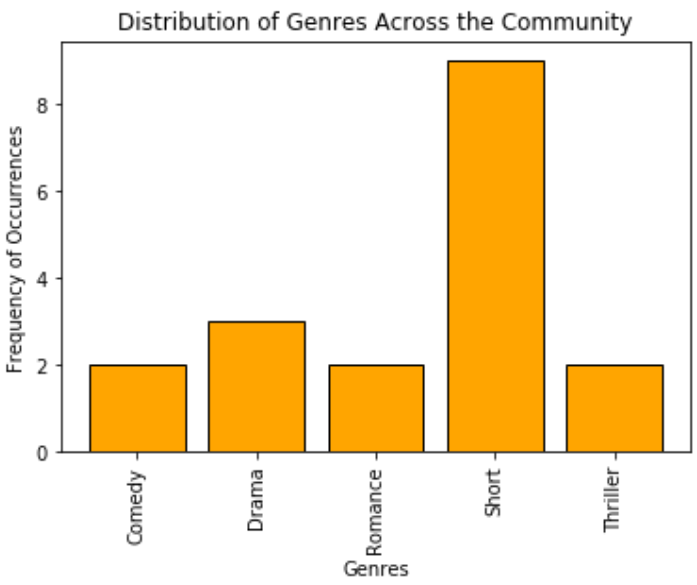
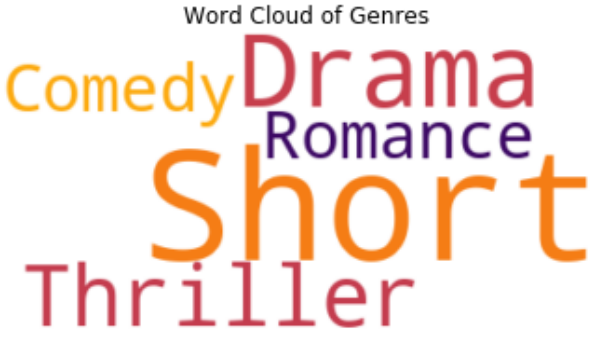


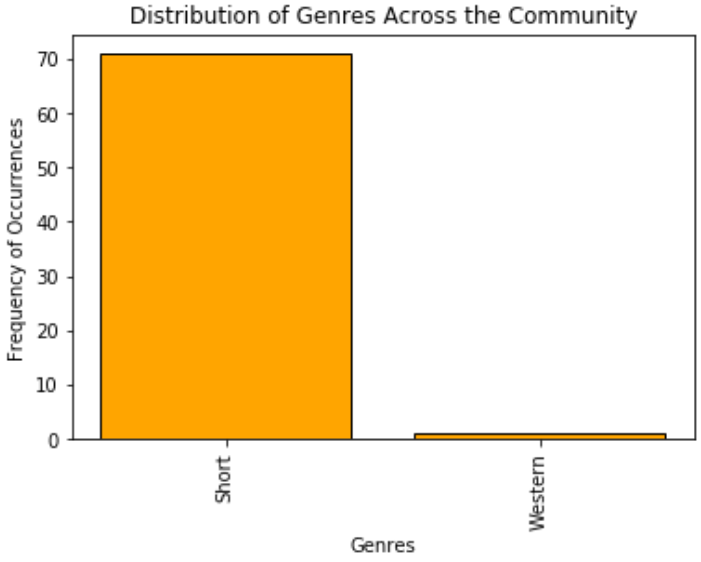


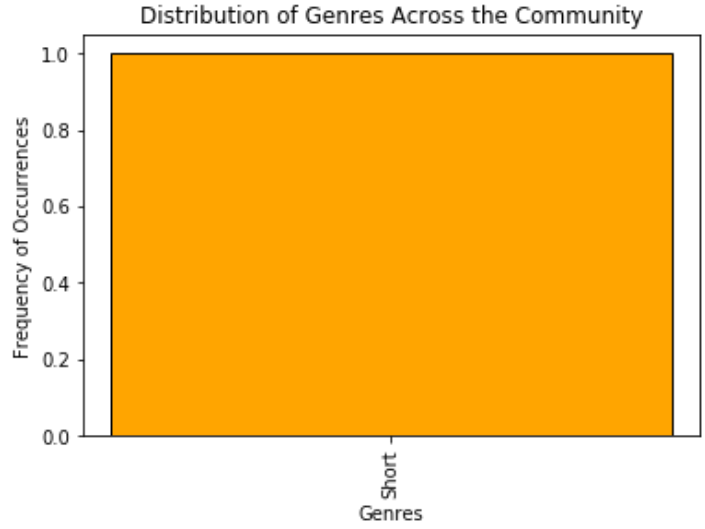


Community # 27
Size of Community: 17
Movies Having Genre Information: 17

Community # 28
Size of Community: 18
Movies Having Genre Information: 18

<table>
<tr>
<td> 
</td>
<td> 
</td>
</tr>
<tr>
<td> 
</td>
<td> 
</td>
</tr>
<tr>
<td>Community # 29
Size of Community: 72
Movies Having Genre Information: 72</td>
<td>Community # 30
Size of Community: 14
Movies Having Genre Information: 1</td>
</tr>
</table>





We observe that the majority of communities have movies of multiple genres present. From the distributions obtained, we can easily find a dominant genre for each community. Also, we see that in nearly every community we have 2-3 communities which have many movies associated with them and the remaining genres have very few occurrences.

We also see that ***Drama*** and ***Short*** are two genres which have a strong presence across all communities.

8)

a) In each community determine the most dominant genre based simply on frequency counts. Which genres tend to be the most frequent dominant ones across communities and why?

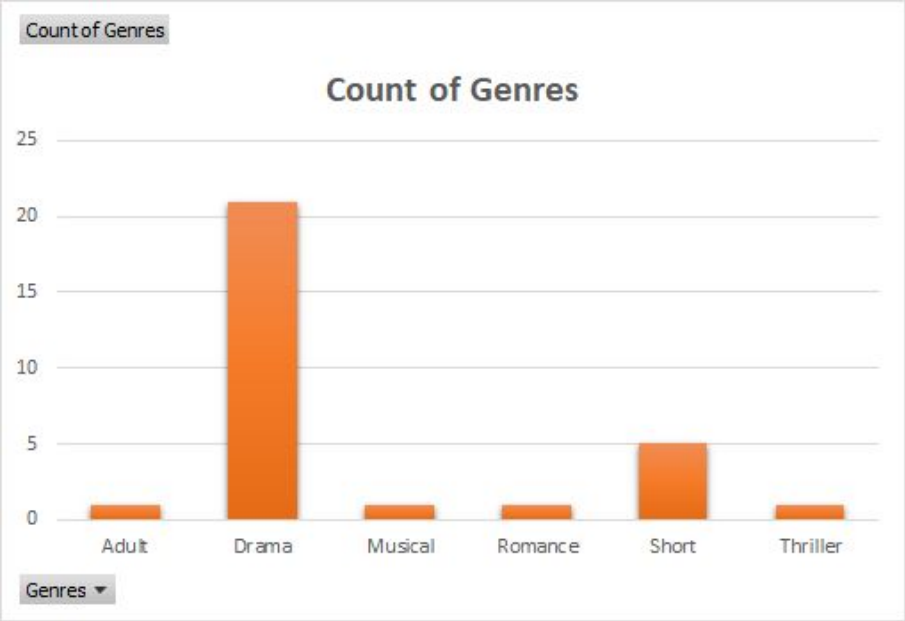
Ans:

In this part, we find the dominant genre in every community. The results obtained are as follows:

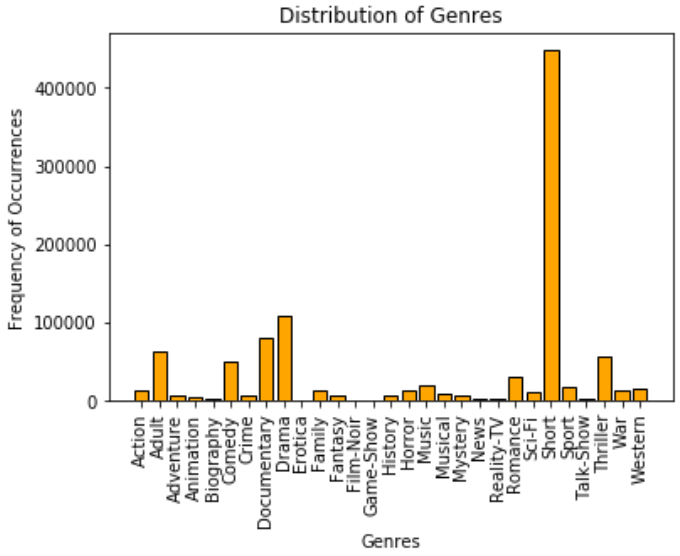
Community Number	Dominant Genre	Top 3 Genres
1	***Romance***	'Romance': 169, 'Drama': 159, 'Comedy': 66
2	***Drama***	'Drama': 9502, 'Thriller': 8635, 'Short': 6547
3	***Adult***	'Adult': 2156, 'Comedy': 457, 'Horror': 423
4	***Drama***	'Drama': 6777, 'Comedy': 4911, 'Thriller': 1736
5	***Drama***	'Drama': 2486, 'Romance': 699, 'Comedy': 659
6	***Short***	'Short': 10906, 'Drama': 6545, 'Western': 5578
7	***Drama***	'Drama': 610, 'War': 257, 'Comedy': 158
8	***Drama***	'Drama': 2602, 'Comedy': 1994, 'Romance': 993
9	***Drama***	'Drama': 1856, 'Comedy': 705, 'Thriller': 693
10	***Drama***	'Drama': 1521, 'Comedy': 913, 'Musical': 305
11	***Drama***	'Drama': 1320, 'Comedy': 716, 'Short': 550
12	***Drama***	'Drama': 672, 'Comedy': 284, 'Romance': 144
13	***Drama***	'Drama': 2453, 'Comedy': 679, 'Romance': 508
14	***Drama***	'Drama': 2658, 'Romance': 2220, 'Thriller': 1334
15	***Drama***	'Drama': 523, 'Comedy': 187, 'Thriller': 146
16	***Drama***	'Drama': 1085, 'Romance': 1022, 'Comedy': 308
17	***Drama***	'Drama': 1779, 'Romance': 960, 'Comedy': 855
18	***Drama***	'Drama': 319, 'Comedy': 255, 'Romance': 117
19	***Drama***	'Drama': 321, 'Romance': 160, 'Thriller': 83
20	***Drama***	'Drama': 692, 'Comedy': 640, 'Romance': 422
21	***Drama***	'Drama': 1636, 'Action': 847, 'Romance': 718
22	***Drama***	'Drama': 8, 'Comedy': 4, 'Thriller': 3

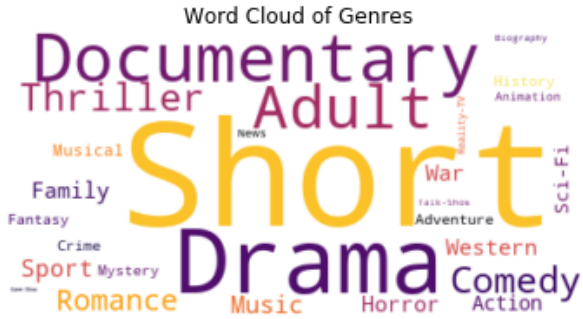
23	***Drama***	'Drama': 104, 'Comedy': 57, 'Action': 29
24	***Thriller***	'Thriller': 11, 'Short': 2, 'Sport': 1
25	***Drama***	'Drama': 240, 'Romance': 45, 'Action': 40
26	***Musical***	'Musical': 75, 'Drama': 38, 'Romance': 18
27	***Short***	'Short': 17
28	***Short***	'Short': 9, 'Drama': 3, 'Thriller': 2
29	***Short***	'Short': 71, 'Western': 1
30	***Short***	'Short': 1

Count of Dominant Genres

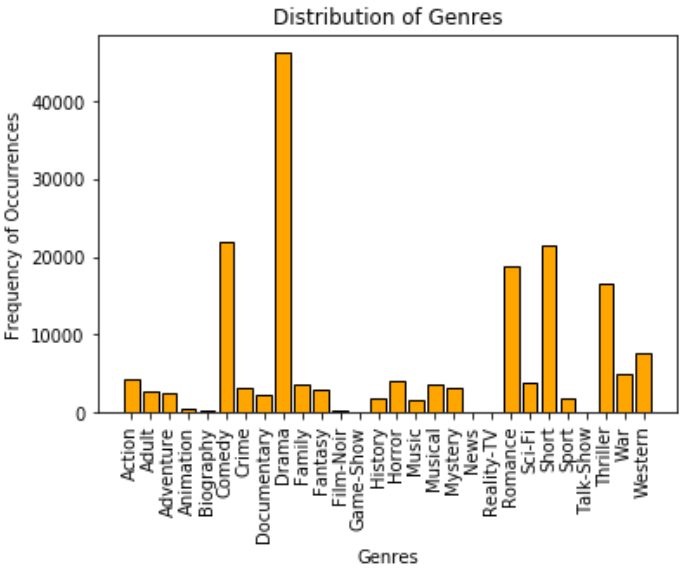


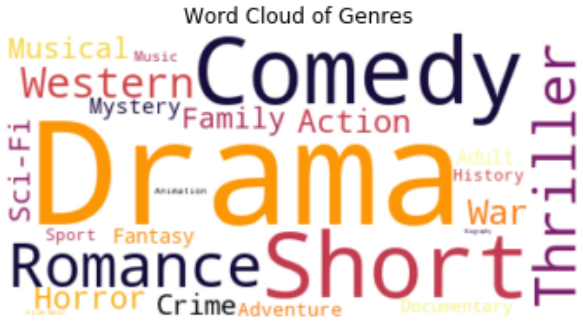
Distribution of Genres Across All Movies (movie_genre.txt)





Distribution of Genres Across All Movies (Movies in the Network)





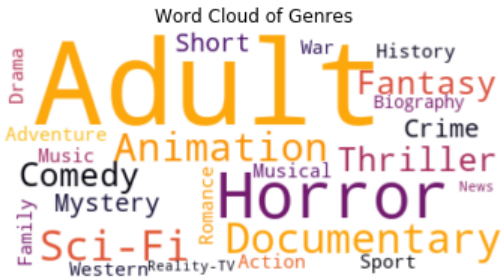
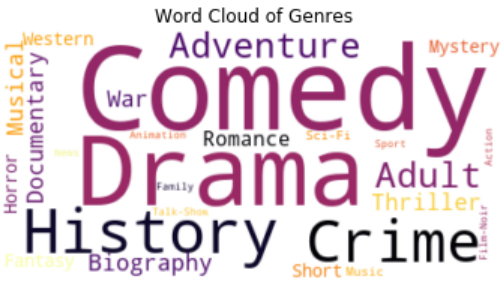
We observe that ***Drama*** and ***Short*** are the most prominent genres appearing across nearly all communities. In the distribution of genres of all movies provided in the genre file, we see that ***Short*** has the maximum number of movies associated with it followed by the ***Drama*** genre. So they have a high probability of appearing across all communities. Following this trend we expect the ***Short*** genre to be the dominant genre across maximum number of communities, but that is not the case - ***Drama*** occurs the maximum times as the dominant genre. This is due to the reason that although the ***Short*** genre occurs for maximum number of movies in the genre file, all movies of

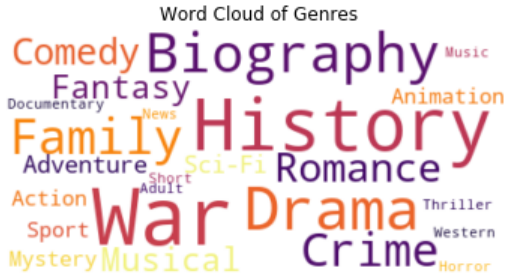
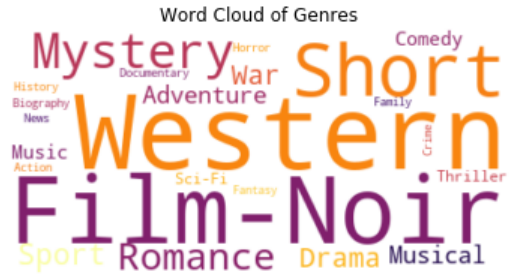
our movie network are not ***Short*** movies - instead, our method of processing and generating movies for the movie network only retains movies with 5 or more actors and hence retains movies of more mainstream genres such as ***Drama, Thriller,*** etc. as they have more number of actors rather than ***Short*** movies which generally contain fewer actors. Hence, we observe the highest occurring mainstream genre, which is ***Drama*** occurring as the dominant genre across most of the communities. This can be further verified by looking at the distribution of the genres of the movies in the movie network graph - we now see that ***Drama*** is the most popular genre and due to its large number it has a significant presence in each community hence being the most dominant genre in most communities.

b) In each community, for the i th genre assign a score of ln(c(i)) * 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. Now determine the most dominant genre in each community based on the modified scores. What are your findings and how do they differ from the results in 8(a).

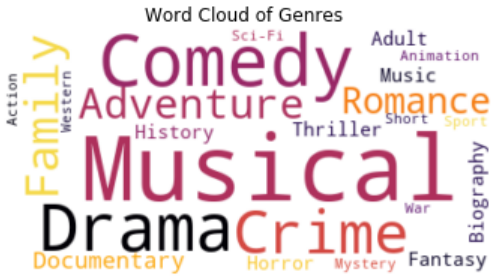
Ans:

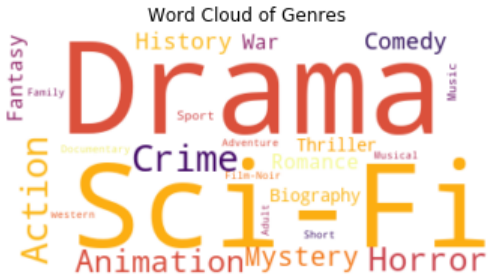
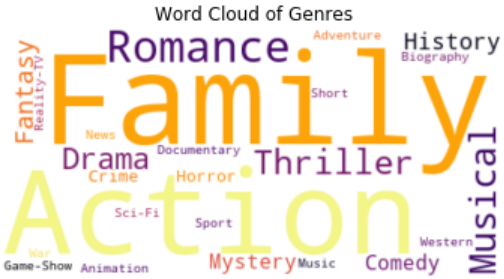
Community Number	**Dominant Genre**	**Scores of Genres**	**Word Cloud**
1	**Romance**	('Romance', 15.632484967382222), ('Drama', 5.927731229648722), ('Crime', 5.477255484013099), ('Comedy', 4.302457634407593), ('Thriller', 3.7414085392470193), ('Mystery', 2.9082300696949526), ('Musical', 2.820802362221907), ('History', 2.6814414794686177), ('War', 2.024585457664956), ('Fantasy', 0.6678605393310223), ('Family', 0.532104280842816), ('Horror', 0.47505005893517965), ('Adventure', 0.4478168909008), ('Sci-Fi', 0.29978121638335226), ('Sport', 0.27464695636633 71), ('Action', 0.11125759750940041), ('Western', 0.0), ('Music', 0.0), ('Documentary', 0.0)	

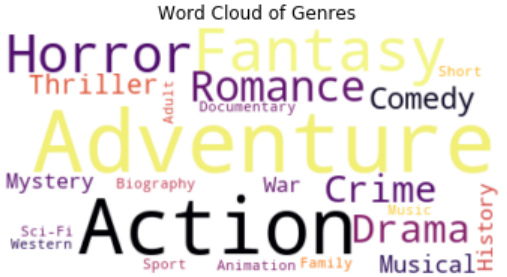
2	**Thriller**	('Thriller', 17.319926354652452), ('Documentary', 17.05166831786052), ('Sci-Fi', 16.38243354927508), ('Animation', 13.443213360130276), ('Horror', 13.184400368856155), ('Sport', 13.064939084067566), ('Music', 12.731932609560335), ('News', 10.929752349004492), ('Short', 9.777651648592856), ('Fantasy', 9.457273802200632), ('Mystery', 7.835446788226868), ('Family', 7.0867140462259695), ('Drama', 6.851358204471368), ('Comedy', 6.580515478734826), ('Romance', 6.525081760896707), ('Crime', 6.451934222456268), ('War', 6.23007904713545), ('Action', 5.818478480579213), ('Game-Show', 5.594992273895155), ('Adventure', 5.025882438991374), ('History', 4.756134651423382), ('Reality-TV', 4.511227307447196), ('Talk-Show', 4.040285139542236), ('Musical', 3.5806071161293938), ('Biography', 2.9974653822415043), ('Western', 1.6498277433722177), ('Adult', 0.41105589347518756), ('Film-Noir', 0.09099741305275307)	
3	**Adult**	('Adult', 230.31350685429612), ('Horror', 24.4527191162446), ('Sci-Fi', 7.872695627059579), ('Documentary', 7.418524125567347), ('Animation', 6.599703162768817), ('Comedy', 4.859487181670817), ('Thriller', 4.809270152987664), ('Fantasy', 4.209137981066273), ('Crime', 2.9457571536283416), ('Mystery', 2.6500745357743956), ('Short', 2.3996837893144156), ('Drama', 1.7842999533706718), ('Romance', 1.0748017525158213), ('Music', 0.931731005393909), ('History', 0.8115597999555968), ('Action', 0.6830348730814297), ('Sport', 0.5830937635581842), ('Western', 0.48600023152138644), ('Musical', 0.4682555518929554), ('War', 0.2526173074065154), ('Adventure', 0.2522130951662954), ('Family', 0.11510877431316628), ('Biography', 0.0), ('Reality-TV', 0.0), ('News', 0.0)	
4	**Comedy**	('Comedy', 15.407939211251788), ('Drama', 10.431513732976171), ('History', 9.992965781772128), ('Crime', 8.394743012390604), ('Adventure', 7.500561271076556), ('Adult', 7.485783578591181), ('Biography', 6.867549280822984), ('War', 6.577189413546014), ('Thriller', 6.3519989498966325), ('Musical', 6.2018761534457685), ('Documentary', 6.1526675764113286), ('Romance', 5.405931981456099), ('Horror', 4.8709915438900415), ('Mystery', 4.6870542191532305), ('Fantasy', 4.641190496379069), ('Western', 4.5500524516798295), ('Short',	

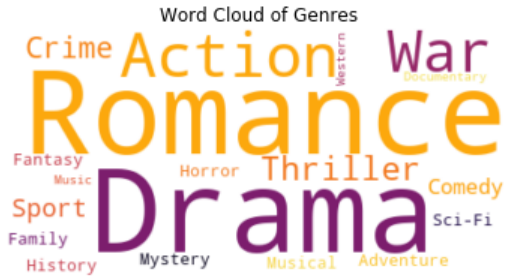
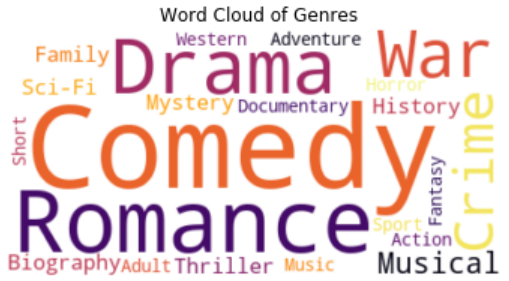
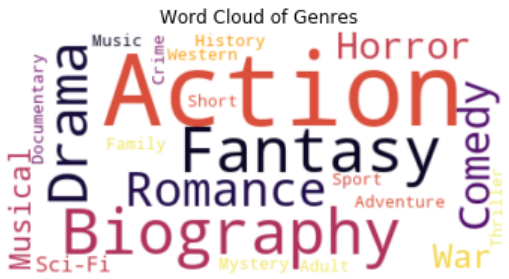
		4.529315440454301), ('Sci-Fi', 2.824472007392786), ('Music', 2.6809062949768374), ('Sport', 1.5219914404355137), ('Family', 1.2291828991866458), ('Animation', 1.0885566071280341), ('Action', 0.6777416034962802), ('News', 0.2603334397286788), ('Film-Noir', 0.050424945533032374), ('Talk-Show', 0.0)	
5	**War**	('War', 22.606018881196576), ('History', 18.635403387567454), ('Drama', 12.120260181207323), ('Biography', 11.680859960047128), ('Family', 9.187181744350962), ('Crime', 8.564805984413784), ('Romance', 6.999093535251151), ('Comedy', 5.642699367004045), ('Fantasy', 4.874677173645546), ('Musical', 4.238548602642863), ('Adventure', 4.225082046539926), ('Sci-Fi', 4.119574303048889), ('Sport', 3.7021280078133625), ('Mystery', 3.5809837080363787), ('Action', 3.3635690702450716), ('Animation', 3.3491650584525265), ('Music', 1.871159232874107), ('Thriller', 1.8354916702533624), ('Short', 1.158960332618527), ('Horror', 0.46438304734866964), ('Documentary', 0.4458894071602623), ('Western', 0.16775691454722724), ('Adult', 0.0), ('News', 0.0)	
6	**Western**	('Western', 32.93720667199386), ('Film-Noir', 27.018441870707996), ('Short', 24.70922169642308), ('Mystery', 9.426536986860247), ('Romance', 7.799765607168337), ('Sport', 7.35033156433687), ('Drama', 6.490961001023162), ('War', 6.127608966124159), ('Musical', 5.754916910861841), ('Adventure', 5.275223822469827), ('Music', 3.894610359915933), ('Comedy', 3.6607463891180325), ('Sci-Fi', 2.576596361629599), ('Thriller', 2.2169281943132897), ('History', 1.8939346787929328), ('Crime', 1.5603315994193316), ('Family', 1.0450962806304813), ('Horror', 0.97234799349637), ('Documentary', 0.8615664848307295), ('Fantasy', 0.8305025145999132), ('Action', 0.5328284341035606), ('Biography', 0.0), ('News', 0.0)	
7	**War**	('War', 34.43627456005908), ('Drama', 10.233330986031008), ('Comedy', 4.426314592934699), ('Family', 3.7848120584627623), ('Romance', 3.721308477526093), ('History', 3.3720442897677114), ('Thriller', 2.5382308484303615), ('Fantasy', 2.3963158775113427), ('Short', 2.1326421366759596), ('Documentary', 2.0724567857627703), ('Horror', 1.125688835559038), ('Crime', 1.1164664659848342), ('Music', 1.094853257364346), ('Musical',	

		1.003206249174255), ('Biography', 0.9988637181076948), ('Sci-Fi', 0.6396954150817171), ('Sport', 0.5669974892991391), ('Animation', 0.4043592882941993), ('Western', 0.3652534263454732), ('Mystery', 0.31554962313440904), ('Action', 0.3068476133784972), ('Adventure', 0.15926415457323922)	
8	**Comedy**	('Comedy', 14.122262182111822), ('Crime', 11.812905449347614), ('Musical', 10.154679338420625), ('Music', 9.689721966835794), ('Family', 9.337511063072089), ('Drama', 9.016153370743163), ('Fantasy', 7.645916106463909), ('Romance', 7.402323801119914), ('History', 6.843595086196503), ('War', 6.336689833031214), ('Adventure', 4.023816486959821), ('Short', 3.6724666444478746), ('Sci-Fi', 3.0675570846452715), ('Documentary', 3.009489495562744), ('Mystery', 2.7478636042196696), ('Reality-TV', 2.5697832446226463), ('Thriller', 2.4210165845789193), ('Biography', 1.653275625927875), ('Sport', 1.585732407119611), ('Horror', 1.499429411399787), ('Animation', 1.1313612860109696), ('Western', 0.5095668028020394), ('Adult', 0.14775668658597843), ('Action', 0.14336950092406178), ('News', 0.0), ('Film-Noir', 0.0)	
9	**Musical**	('Musical', 13.921049678076676), ('Western', 13.68153622492784), ('Action', 12.241046834430136), ('Mystery', 9.675204139927022), ('Drama', 9.157774298746514), ('Thriller', 8.354094835232296), ('Comedy', 6.412511980159293), ('Family', 5.714741610378057), ('Fantasy', 5.064537851008965), ('Romance', 4.971135045174466), ('War', 4.29508373267663), ('Sci-Fi', 3.733743882413801), ('Horror', 2.5133854316929467), ('History', 2.2637050220626547), ('Crime', 1.8817474837271786), ('Adventure', 1.6181330650536583), ('Music', 1.4528189331986032), ('Sport', 0.7173439391897771), ('Documentary', 0.6771414239646305), ('Short', 0.5191415927650082), ('Animation', 0.4053871461698836), ('Biography', 0.0)	

10	**Musical**	('Musical', 22.202908051484233), ('Comedy', 13.027187476315643), ('Drama', 11.026787274127582), ('Crime', 9.378705184301626), ('Family', 5.728299112009687), ('Adventure', 4.959636458600404), ('Romance', 3.491147794286465), ('Documentary', 3.431586385310777), ('Adult', 2.7135813189815527), ('Thriller', 2.668920690418559), ('History', 2.586043091065892), ('Music', 2.318990657527529), ('Horror', 2.2542616326598606), ('Biography', 2.1932413751445425), ('Fantasy', 2.164704711121293), ('Mystery', 1.4484186927285154), ('Sci-Fi', 0.9846727713102168), ('Western', 0.941933466168498), ('Action', 0.8688152582510802), ('War', 0.6586999272151366), ('Sport', 0.6136828819924094), ('Short', 0.35569177434948573), ('Animation', 0.15295329600693625)	
11	**Family**	('Family', 32.89429232159855), ('Crime', 10.021775135449062), ('Comedy', 9.275553554520505), ('Drama', 8.835379439915352), ('Short', 6.979904877563147), ('Romance', 3.9078739494537924), ('Thriller', 3.700263871311253), ('War', 3.014627119241187), ('Music', 2.920750268544687), ('Documentary', 2.4857934199757663), ('Fantasy', 2.3946627005450383), ('Mystery', 2.3921398357085994), ('Animation', 2.054164253302578), ('Musical', 1.9377602540946355), ('History', 1.901707870487201), ('Biography', 1.422893753027961), ('Sci-Fi', 1.2263738730202303), ('Sport', 1.2085687120430753), ('Horror', 0.9191471346688646), ('Adventure', 0.5738266620531202), ('Action', 0.16909654598587415), ('Western', 0.1490073700238583)	
12	**History**	('History', 20.617472611098094), ('Drama', 10.239003286550794), ('Comedy', 7.943195587322952), ('War', 6.652662962643957), ('Animation', 4.341541832211403), ('Romance', 4.106274529861979), ('Fantasy', 3.910536259983825), ('Crime', 3.7153105688580936), ('Musical', 3.204458055677162), ('Family', 3.1156401101534508), ('Adventure', 2.590482409248748), ('Short', 2.270463333778696), ('Sci-Fi', 1.8505041976235013), ('Thriller', 1.7511746911604738), ('Music', 1.4221564316421456), ('Action', 1.0373733370967488), ('Sport', 0.8587229059200412), ('Mystery', 0.6938022630557404), ('Documentary', 0.6836357517936194), ('Horror', 0.6277005476375959), ('Western', 0.5243856405830305)	

13	**Drama**	('Drama', 12.57967511583563), ('Sci-Fi', 10.16318956227273), ('Crime', 9.729479857069721), ('Action', 9.616588164497083), ('Horror', 9.081247354562716), ('Animation', 8.784518619371603), ('Mystery', 8.358372121980743), ('History', 6.259719973869576), ('Comedy', 6.15419305138762), ('Fantasy', 5.686043224395198), ('War', 5.184166713031884), ('Romance', 5.098422087386558), ('Thriller', 4.284838345577999), ('Biography', 4.167415888083684), ('Music', 2.9791331844247653), ('Sport', 2.8624121919453756), ('Musical', 1.7363466728869543), ('Adventure', 1.4553194519118626), ('Family', 1.3864178589687497), ('Short', 0.9898575618532933), ('Documentary', 0.6250495269514222), ('Adult', 0.6233228673795417), ('Western', 0.09174759129839587), ('Film-Noir', 0.0)	
14	**Family**	('Family', 26.82176715563693), ('Action', 22.906823977147653), ('Romance', 17.02416343313695), ('Musical', 11.532421170080644), ('Thriller', 10.954585143012174), ('Drama', 8.50834217963709), ('Fantasy', 5.438420842724576), ('History', 4.400246414203286), ('Mystery', 3.2857419297611212), ('Comedy', 3.194720023160904), ('Crime', 2.872379421182383), ('Horror', 2.661824920583299), ('Sport', 1.311045624979808), ('War', 1.0948345674194437), ('Adventure', 0.7652101066739856), ('Sci-Fi', 0.5792685702304933), ('Music', 0.5533702409321067), ('Animation', 0.486651648494218), ('Biography', 0.3685460512901746), ('Documentary', 0.25958238648443216), ('Western', 0.04868227227106487), ('Short', 0.04221436981733615), ('Reality-TV', 0.0), ('News', 0.0), ('Game-Show', 0.0)	
15	**Family**	('Family', 15.73730191066573), ('Drama', 8.471666140639984), ('Comedy', 5.355183135208805), ('Thriller', 5.292989804645685), ('War', 4.672623024182652), ('Romance', 4.426827218181374), ('Sport', 4.17634991817229), ('Action', 4.034473647939288), ('Horror', 4.027057396336391), ('Music', 2.6514547515196973), ('Mystery', 2.6380198918711306), ('History', 2.5589919043415934), ('Short', 2.479243675792394), ('Biography', 2.3493270025387045), ('Crime', 2.219291419554591), ('Fantasy', 2.2046573341623175), ('Adventure', 1.9419018907477212), ('Documentary', 1.6547291927736711), ('Musical', 1.4764927569387236), ('Sci-Fi', 0.954277284341417), ('Animation', 0.4000316972489102), ('Western', 0.12628443428954392), ('Adult',	

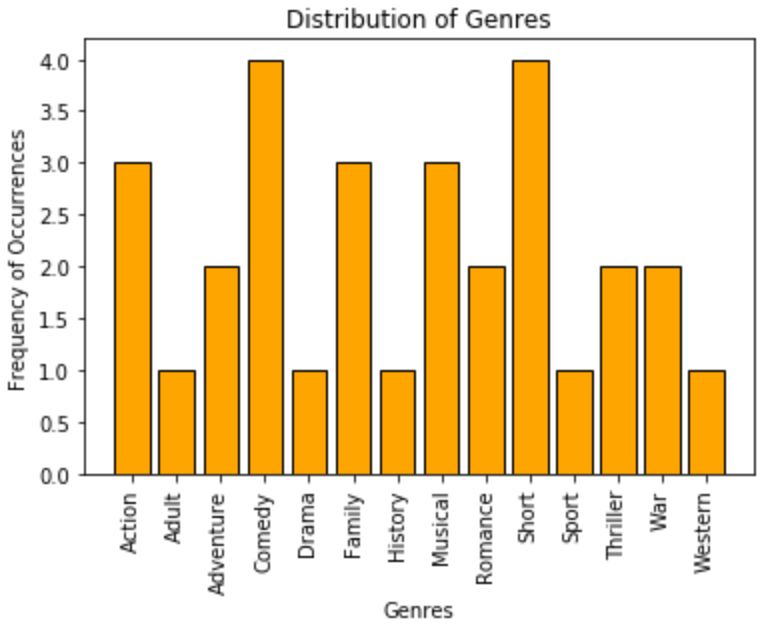
		0.060832962388529775)	
16	**Adventure**	('Adventure', 37.217130923750226), ('Romance', 21.016739640782763), ('Drama', 9.179560344698816), ('Crime', 6.732010714561219), ('Comedy', 4.519457496551073), ('History', 3.8844729525279402), ('War', 3.7973174102891316), ('Fantasy', 3.780055429088326), ('Western', 1.508263340635231), ('Family', 1.4734993804379233), ('Action', 1.2323749828845352), ('Biography', 1.0989498963690125), ('Sci-Fi', 0.8969137408309099), ('Thriller', 0.8707213015920324), ('Musical', 0.7498743660776797), ('Mystery', 0.5410982902804835), ('Horror', 0.1134611375930953), ('Sport', 0.10746425063445522), ('Documentary', 0.08555309665894072), ('Music', 0.05156463544355227), ('Short', 0.014492178963887228), ('Animation', 0.0)	
17	**Adventure**	('Adventure', 27.939830113045122), ('Action', 22.5797887660425), ('Fantasy', 12.055421434979332), ('Horror', 10.838948974104255), ('Romance', 10.08139353239), ('Drama', 8.304811650825233), ('Crime', 8.015123200612859), ('Comedy', 7.617091351234979), ('Thriller', 5.3098115587163), ('Musical', 4.423999244199219), ('Mystery', 3.890074563858223), ('War', 3.208410185157768), ('History', 3.1023332118675904), ('Sci-Fi', 1.4710978178691834), ('Sport', 1.074265899008629), ('Biography', 0.9528019600653211), ('Documentary', 0.9096465692311757), ('Family', 0.689402070603634), ('Animation', 0.385712600701142), ('Music', 0.20606415970132236), ('Western', 0.07480523294796003), ('Short', 0.035523532491618416), ('Adult', 0.0)	
18	**Comedy**	('Comedy', 11.01386789473005), ('Music', 8.164842404576152), ('Drama', 6.776073872651596), ('Musical', 5.6817668378747195), ('Romance', 5.032920671315096), ('War', 5.018751456114842), ('Thriller', 3.5197310492958107), ('Family', 3.4666627181538434), ('Short', 2.660705456526626), ('History', 2.601158651809753), ('Documentary', 2.3560191839967985), ('Fantasy', 2.229172317897379), ('Crime', 1.7585526724121934), ('Sport', 1.6510205605564952), ('Horror', 0.7139323742854413), ('Sci-Fi', 0.3666733843640903), ('Western', 0.3043535053470684), ('Mystery', 0.18204049854462523), ('Adventure', 0.0), ('Biography', 0.0), ('Animation', 0.0)	

19	**Romance**	('Romance', 9.724426328501451), ('Drama', 9.049553684099454), ('Action', 7.86482816313896), ('War', 6.4427823763420236), ('Thriller', 5.036245441515593), ('Crime', 4.754720602357568), ('Sport', 3.0296718037756567), ('Comedy', 2.6442060466100137), ('Horror', 1.8971717010922382), ('History', 1.7808563361117336), ('Mystery', 1.6860952895648131), ('Adventure', 0.7261720974742167), ('Fantasy', 0.4435538430405527), ('Family', 0.3533924895496481), ('Musical', 0.34838331307187376), ('Sci-Fi', 0.33497679327434576), ('Documentary', 0.14521352646346175), ('Music', 0.0), ('Western', 0.0)	
20	**Comedy**	('Comedy', 16.093558640020664), ('Romance', 11.505040029777403), ('Drama', 8.324920877675538), ('War', 6.784153891650306), ('Crime', 4.133647708449048), ('Musical', 2.836830803503783), ('Thriller', 1.7573550174800767), ('Mystery', 1.2233698422558243), ('History', 1.133730055919501), ('Biography', 0.7024819016078367), ('Sci-Fi', 0.6783854208703844), ('Family', 0.5526414622737009), ('Documentary', 0.5373513541202344), ('Adventure', 0.4629158571622618), ('Music', 0.4548919220897192), ('Sport', 0.39875857663976544), ('Western', 0.15195966743434228), ('Short', 0.14674475882587232), ('Fantasy', 0.09928474268007607), ('Horror', 0.07062136491666121), ('Action', 0.06615861419804173), ('Adult', 0.043245496324703765)	
21	**Action**	('Action', 49.38602083771655), ('Biography', 12.214236010750861), ('Fantasy', 12.051714992967101), ('Drama', 9.595548618685154), ('Romance', 9.175436210512071), ('Comedy', 7.386046000555899), ('Horror', 7.105556375656342), ('War', 5.323758361646138), ('Musical', 4.58058140850857 2), ('Sci-Fi', 2.0625949754604784), ('Thriller', 1.9590097529749273), ('Sport', 1.899878122464333), ('Crime', 0.9129730483781535), ('Adventure', 0.8772930593091948), ('Family', 0.8759827170423559), ('History', 0.7837928492093184), ('Music', 0.6424174876575126), ('Mystery', 0.35430242485892754), ('Western', 0.3466345247893515), ('Documentary', 0.18272541133397543), ('Adult', 0.10815836809844523), ('Short', 0.07591395856354613)	

22	**Sport**	('Sport', 14.932858479466038), ('Drama', 2.7981558960060555), ('Comedy', 1.9731849203729104), ('Thriller', 1.5584164055231842), ('War', 0.0), ('Fantasy', 0.0), ('Romance', 0.0), ('Western', 0.0), ('Crime', 0.0)	
23	**Action**	('Action', 13.78694512827159), ('History', 9.134503626756658), ('Comedy', 6.308020434944089), ('Drama', 6.24962496771655), ('War', 6.214204694137949), ('Mystery', 5.783779718182025), ('Crime', 2.5227381086440857), ('Romance', 1.9005533903952094), ('Thriller', 1.8923465029591509), ('Family', 1.35843884700241), ('Musical', 0.9228079730867021), ('Documentary', 0.9144739027490788), ('Fantasy', 0.6983137587163879), ('Music', 0.5511724912294083), ('Sci-Fi', 0.22182409721846305), ('Biography', 0.0), ('Adventure', 0.0), ('Horror', 0.0), ('Short', 0.0)	
24	**Thriller**	('Thriller', 20.48993124610799), ('Short', 0.8270893565818496), ('Sport', 0.0)	
25	**Action**	('Action', 16.09546228735415), ('Drama', 13.149053963851848), ('Fantasy', 6.048141947783585), ('Family', 5.2979036280635645), ('Musical', 4.287244585639778), ('Romance', 4.198210304004039), ('History', 1.483671407293757), ('Crime', 0.4716037775472905), ('Comedy', 0.3518081640974956), ('Mystery', 0.20774764036369497), ('Horror', 0.0), ('Thriller', 0.0), ('Sci-Fi', 0.0)	

26	**Musical**	('Musical', 104.62579514609294), ('Biography', 9.592983370657667), ('Action', 4.560116594021658), ('Drama', 3.47247561689575), ('Romance', 3.204230559861058), ('Thriller', 0.5682888709081074), ('Horror', 0.4056433944803644), ('Comedy', 0.2946964491466035), ('Crime', 0.0), ('Sport', 0.0)	
27	**Short**	('Short', 23.66487907995374)	
28	**Short**	('Short', 9.176339304494745), ('Thriller', 0.8375844225949834), ('Romance', 0.7304680422101407), ('Drama', 0.7083636099017523), ('Comedy', 0.6303229606746796)	
29	**Short**	('Short', 35.110222540853734), ('Western', 0.0)	

30	**Short**	('Short', 0.0)	Word Cloud of Genres Short



In the previous section, we saw that **Drama** was the most dominant genre based on simple frequency count and it was dominant by a high margin as well. With this new method of finding the score, things have changed significantly with Drama no longer being a dominant genre. Now different communities have different genres as the dominant genre.

As this is a Data Science project, just number are not relevant and we investigated a bit further to understand this change. Let us consider a simple example to understand how the factor p(i)/q(i) affects the score. Consider just 3 communities A, B & C with the following genre distribution based on count.

Genre	A	B	C

Drama	10	4	3
Comedy	0	3	0
Others	10	0	2
Total	20	7	5

Based on the frequency counts as per 8(a), the dominant genre is Drama in all 3 communities. Now as per 8(b), the dominant genres for B becomes Comedy as shown below.:

Drama: p(i)/q(i) = 1.076

Score = ln(4) * (4/7) / (17/32) = 1.484

Comedy: p(i)/q(i) = 4.5

Score = ln(3) * (3/7) / (3/32) = 4.98

As seen above, the factor p(i)/q(i) captures the essence of how close the local distribution of genre in the community is to its global distribution. And when this factor is high, it denotes that there is higher concentration of that genre is in that community compared to its global distribution. This is helpful in identifying the dominant genre considering the global information as well.

c) Find a community of movies that has size between 10 and 20. Determine all the actors who acted in these movies and plot the 5 corresponding bipartite graph (i.e. restricted to these particular movies and actors). Determine three most important actors and explain how they help form the community. Is there a correlation between these actors and the dominant genres you found for this community in 8(a) and 8(b).

Ans:

In this problem, we explore two communities with size between 10 and 20 and try to understand the correlation between the actors acted in these movies and the dominant genres obtained in the previous questions.

1. **Community Number 27**
    - This community consists of 17 movies
    - On looking up these movies on the IMDB website, we find out that all these movies are Norwegian Short films.
    - There are 11 actors associated with these movies

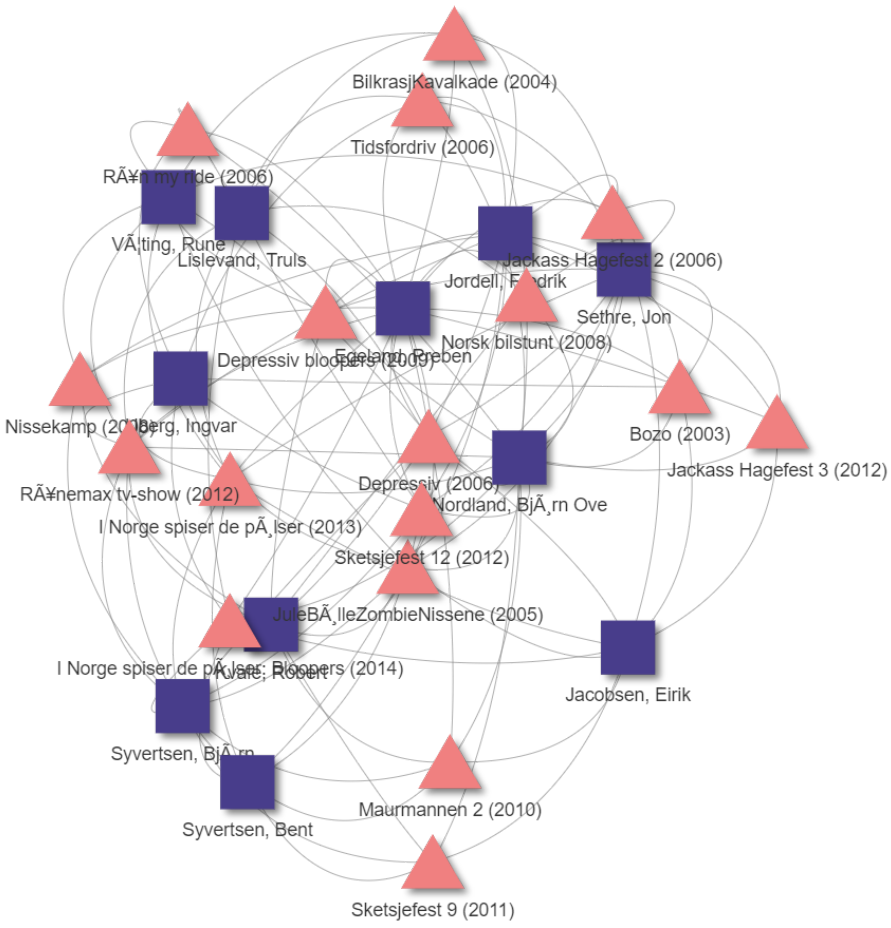
- The details of the movies and the actors acted in this community is given below:

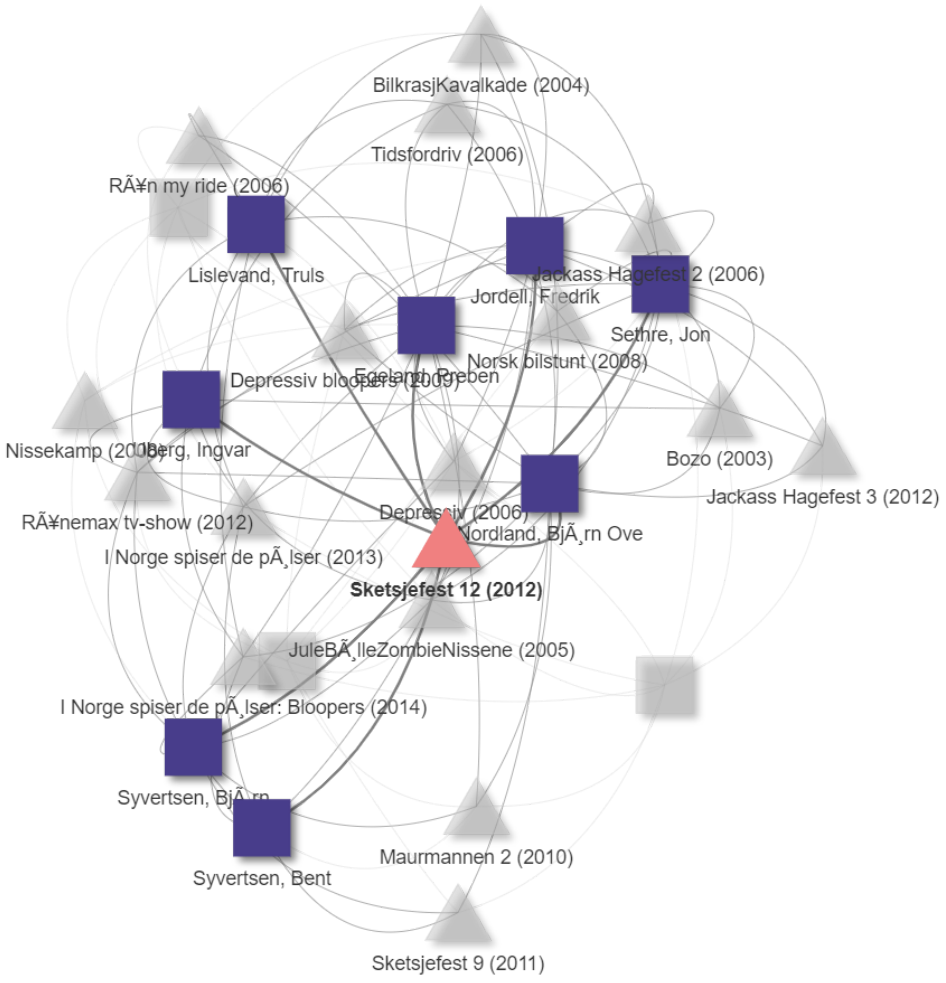
Community Number 27	
Movies	**Actors**
Nissekamp (2008)	Syvertsen, Bent
Depressiv (2006)	Uberg, Ingvar
Rån my ride (2006)	Egeland, Preben
Maurmannen 2 (2010)	Jordell, Fredrik
Jackass Hagefest 3 (2012)	Væting, Rune
I Norge spiser de pølser: Bloopers (2014)	Nordland, Bjørn Ove
I Norge spiser de pølser (2013)	Lislevand, Truls
Tidsfordriv (2006)	Sethre, Jon
Depressiv bloopers (2009)	Kvale, Robert
Norsk bilstunt (2008)	Syvertsen, Bjørn
JuleBølleZombieNissene (2005)	Jacobsen, Eirik
Jackass Hagefest 2 (2006)	
Rånemax tv-show (2012)	
Sketsjefest 12 (2012)	
BilkrasjKavalkade (2004)	
Bozo (2003)	
Sketsjefest 9 (2011)	

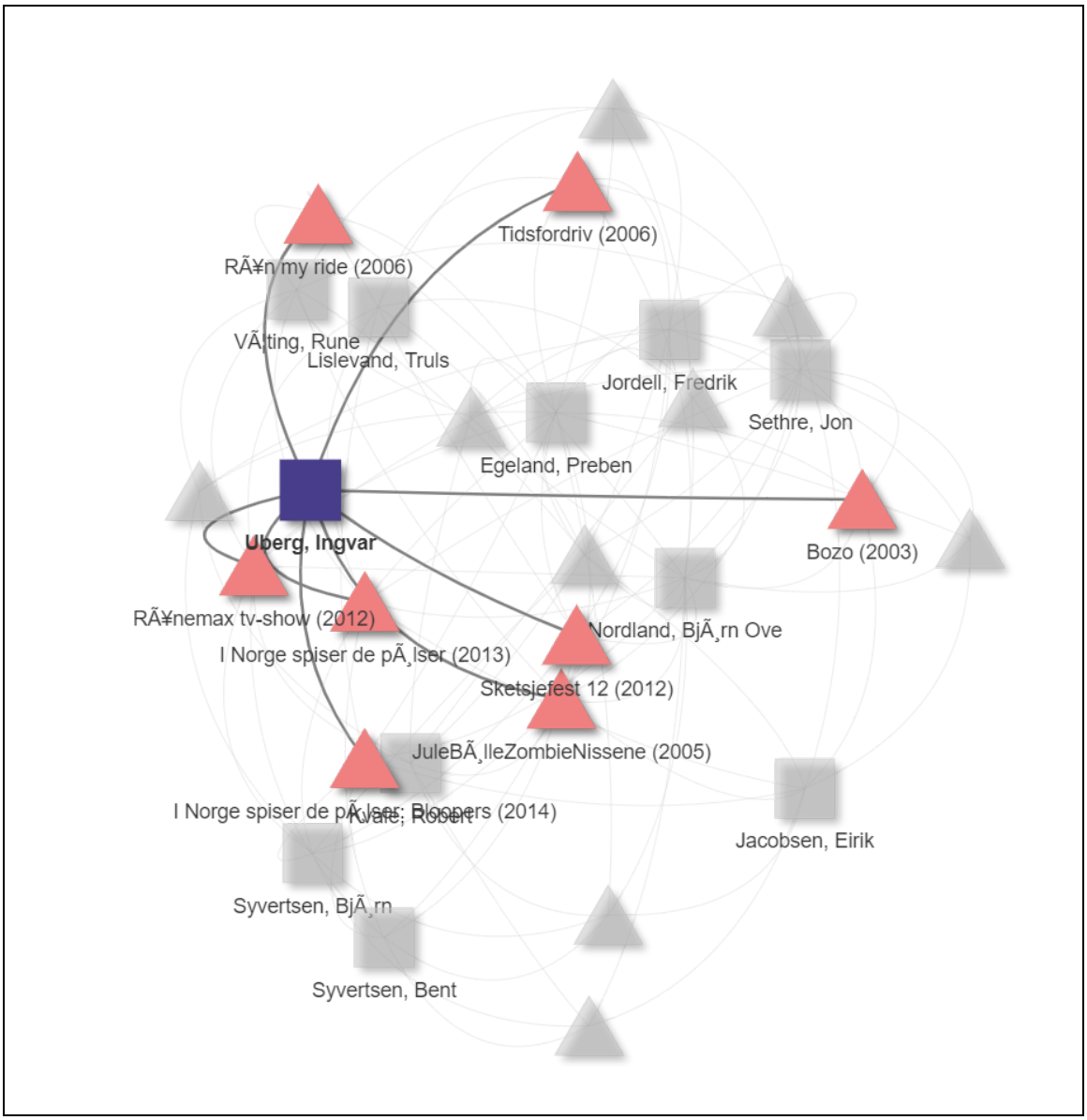
Number of Movies in the Community	17
Unique Actors Related to the Above Movies	11
Edges Created	113
Dominant Genre from 8(a)	Short

Dominant Genre from 8(b)	Short

- The bipartite graph created for this network is shown in the figures in the table.
- We have represented the movies in the community as red triangles and the actors who have acted in them as purple squares.
- We have made use of visnetwork to better visualize the bipartite network.
- The first image in the table shows the network in its entirety; the second image shows the graph when a movie is selected - only its associated actors get highlighted; and the third image shows how the graph looks when an actor is selected.

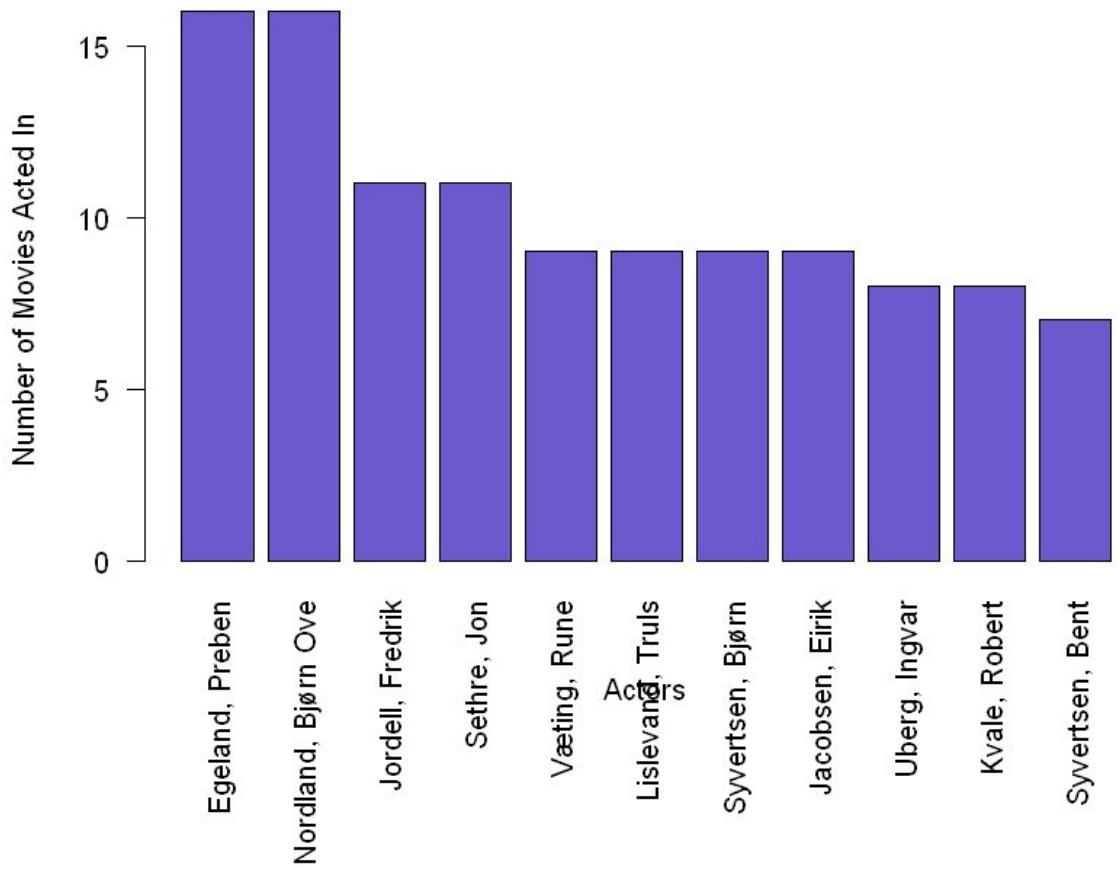
Bipartite Graph Created


Movie Node

Actor Node

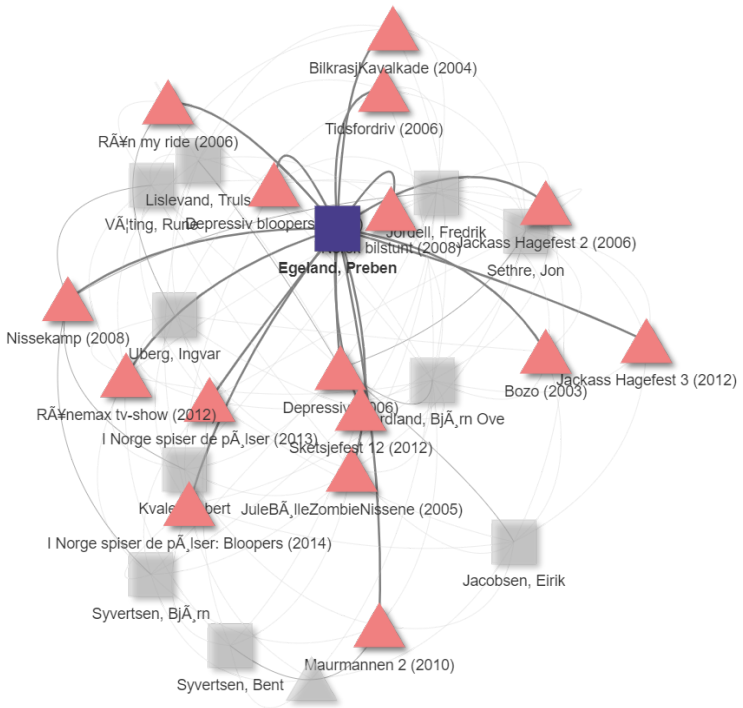


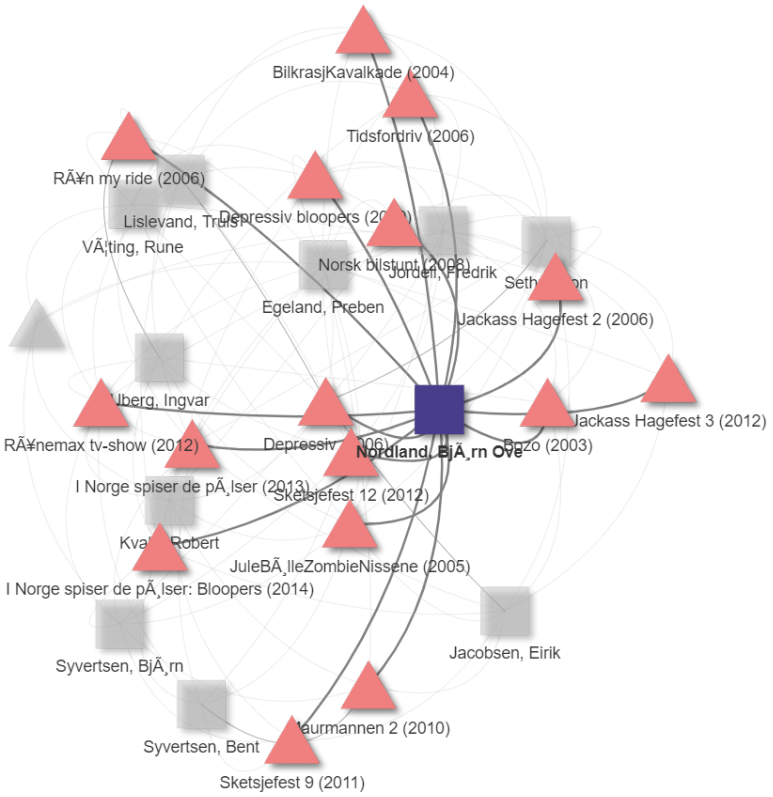
- Next, we find the four most important actors in these communities. We do that by inspecting the in degree of all the actor nodes and determining the actors with the maximum in degree as the most important actors for the community.
- The in degrees of the actors is shown below. On the basis of this, the most important actors are:
    - **Egeland, Preben**
    - **Nordland, Bjørn Ove**
    - **Jordell, Fredrik**
    - **Sethre, Jon**

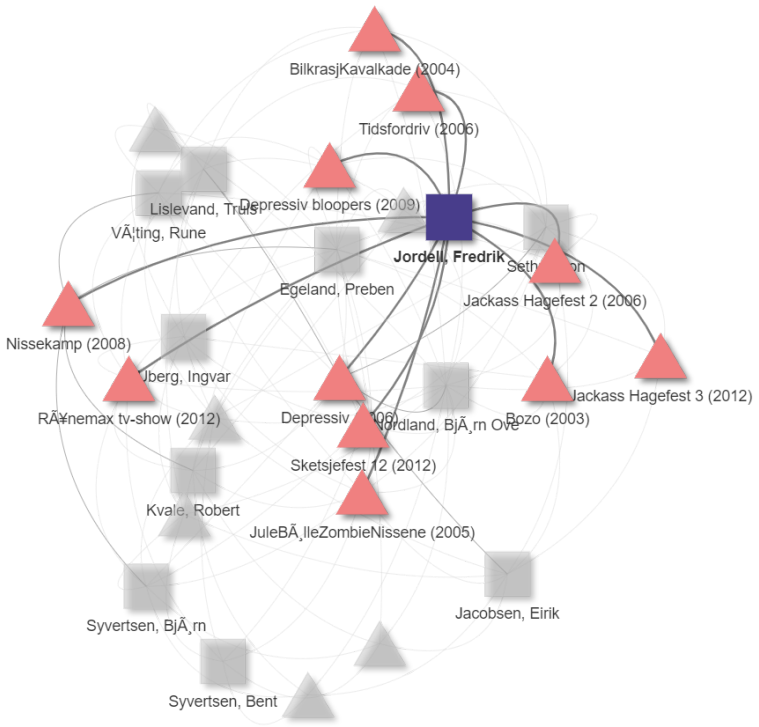
Number of Movies in the Community Associated with the Actors

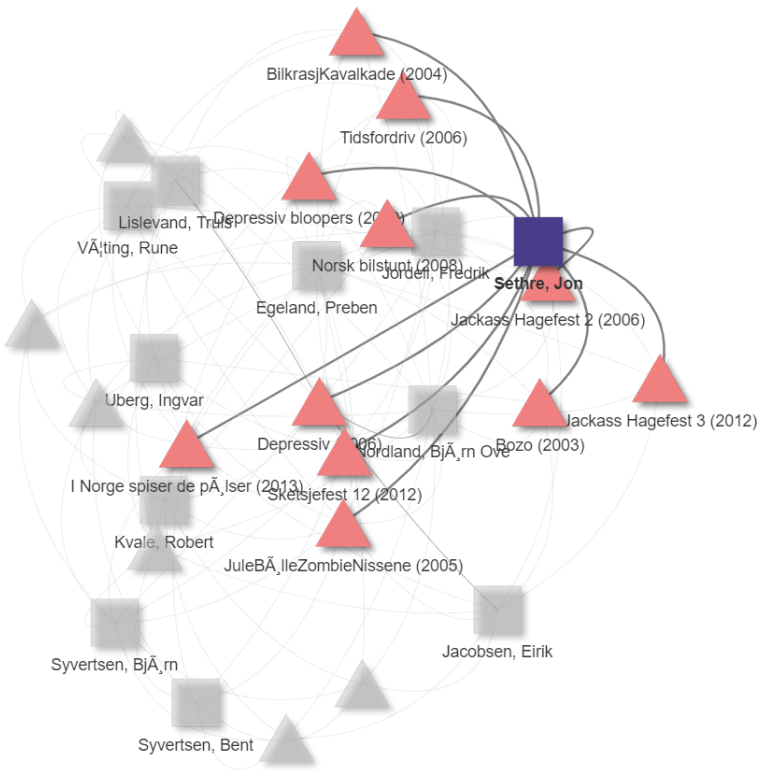


- We further analyse these top actors locally as well as globally (considering all their movies) and obtain the following statistics:

Details of the Top 4 Actors		
Egeland, Preben	***Number of Movies Acted in in this Community***	16
	Visualization of the Movies Acted in in this Community	
	Total Number of Movies Acted in	68
	Dominant Genre Across all Movies	'Short': 68
Nordland, Bjørn Ove	***Number of Movies Acted in in this Community***	16

	Visualization of the Movies Acted in in this Community	
	Total Number of Movies Acted in	46
	Dominant Genre Across all Movies	'Short': 46
Jordell, Fredrik	***Number of Movies Acted in in this Community***	11

	Visualization of the Movies Acted in in this Community	
	Total Number of Movies Acted in	14
	Dominant Genre Across all Movies	'Short': 14
Sethre, Jon	***Number of Movies Acted in in this Community***	11

	Visualization of the Movies Acted in in this Community	
	Total Number of Movies Acted in	43
	Dominant Genre Across all Movies	'Short': 43

- We observe that for all the top actors, all the movies that they have acted in across the entire network are ***Short*** movies. This corresponds to the dominant genre obtained for this community from both question 8(a) and 8(b).
- From these findings, we observe that there is a possible correlation between these actors and the dominant genres obtained, i.e., actors tend to have a particular genre associated with them derived from the genres of movies they act in and when there are movies which feature a lot of actors which have the same genre associated with them, they tend to form communities and the dominant genre for the community will be the dominant genre for all the actors present as can be observed for this community.

2. Community Number 28

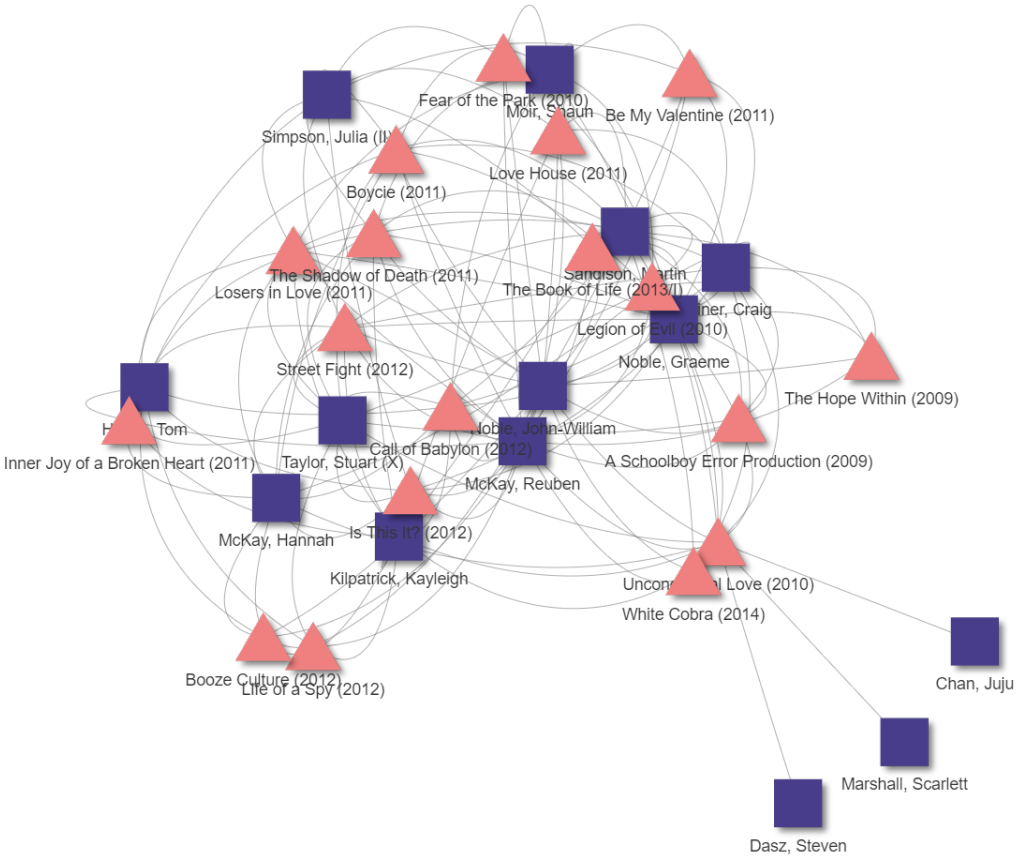
- This community consists of 18 movies.
- This community consists of movies from various genres such as ***short, comedy, drama, romance,*** and ***thriller***.
- There are 14 actors associated with these movies
- The details of the movies and the actors acted in this community is given below:

Community Number 28	
Movies	**Actors**
Street Fight (2012)	Sandison, Martin
Inner Joy of a Broken Heart (2011)	Taylor, Stuart (X)
Is This It? (2012)	McKay, Hannah
Call of Babylon (2012)	Marshall, Scarlett
Life of a Spy (2012)	Simpson, Julia (II)
Booze Culture (2012)	Kilpatrick, Kayleigh
Losers in Love (2011)	Moir, Shaun
Boycie (2011)	Chan, Juju
Unconditional Love (2010)	Noble, John-William
The Shadow of Death (2011)	McKay, Reuben
Be My Valentine (2011)	Dasz, Steven
Legion of Evil (2010)	Hislop, Tom
Love House (2011)	Joiner, Craig
The Hope Within (2009)	Noble, Graeme
White Cobra (2014)	
A Schoolboy Error Production (2009)	
The Book of Life (2013/I)	
Fear of the Park (2010)	

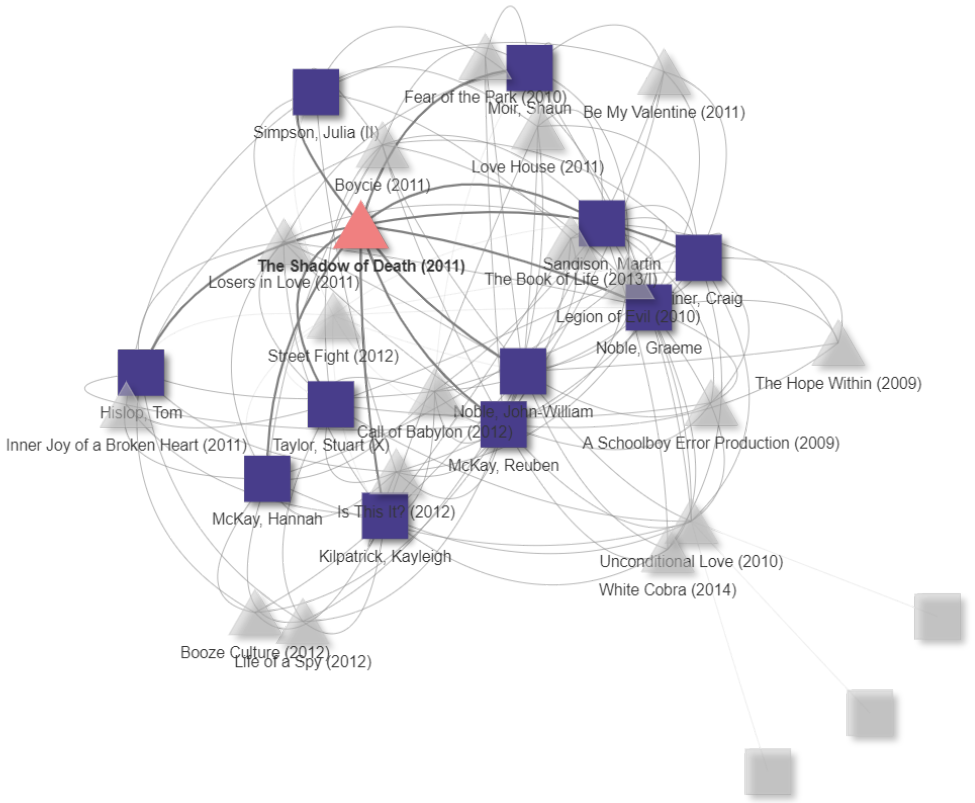
Number of Movies in the Community	18
Unique Actors Related to the Above Movies	14
Number of Edges	138
Dominant Genre from 8(a)	Short
Dominant Genre from 8(b)	Short

- The bipartite graph created for this network is shown in the figures in the table.
- We have represented the movies in the community as red triangles and the actors who have acted in them as purple squares.
- We have made use of visnetwork to better visualize the bipartite network.
- The first image in the table shows the network in its entirety; the second image shows the graph when a movie is selected - only its associated actors get highlighted; and the third image shows how the graph looks when an actor is selected.

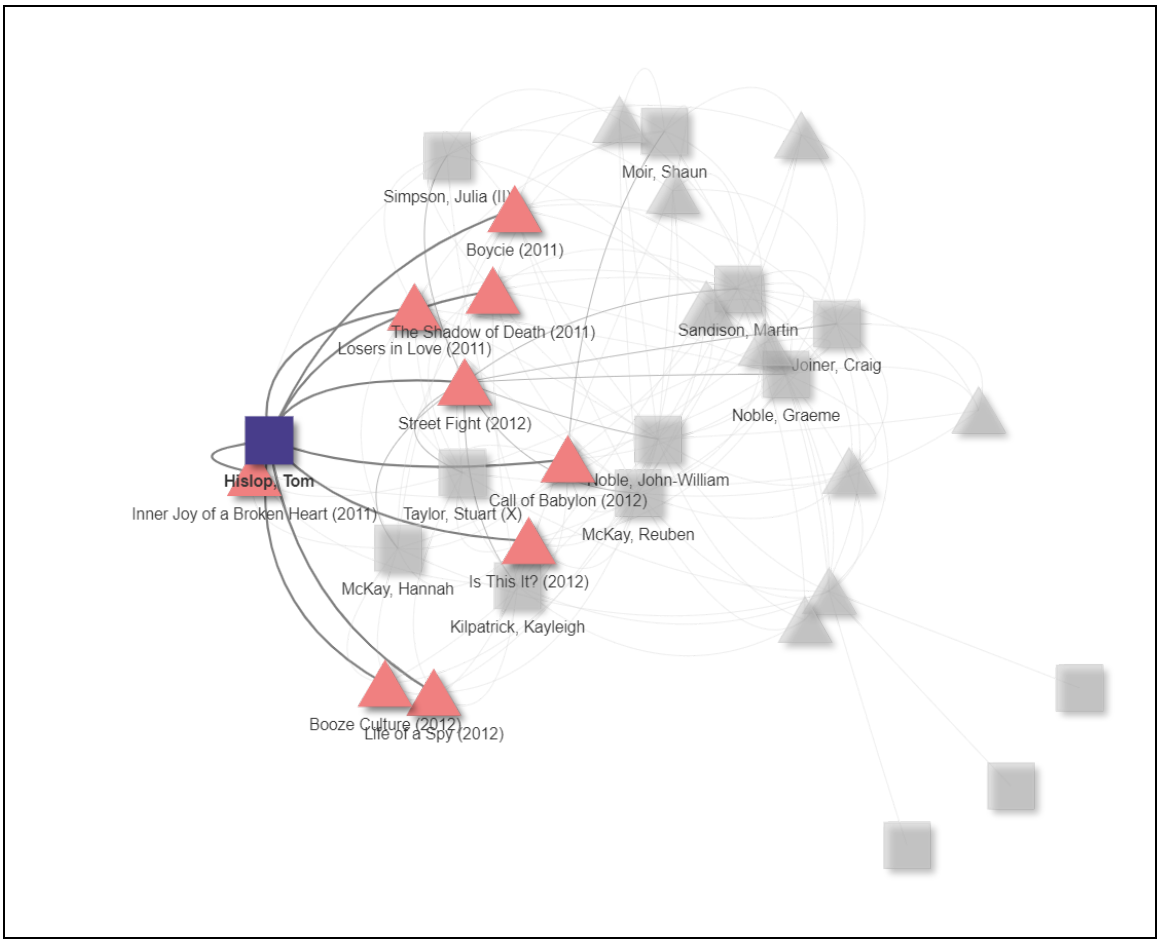
Bipartite Graph Created


Fear of the Park (2010)
Be My Valentine (2011)
Simpson, Julia (II
Love House (2011)
Boycie (2011)
The Shadow of Death (2011)
Losers in Love (2011)
The Book of Life (2013/I)
Legion of Evil (2010)
Noble, Graeme
Street Fight (2012)
The Hope Within (2009)
Inner Joy of a Broken Heart (2011)
Taylor, Stuart (X)
Call of Babylon (2012)
A Schoolboy Error Production (2009)
McKay, Reuben
McKay, Hannah
Is This It? (2012)
Kilpatrick, Kayleigh
White Cobra (2014)
Booze Culture (2012)
Life of a Spy (2012)
Chan, Juju
Marshall, Scarlett
Dasz, Steven

Movie Node

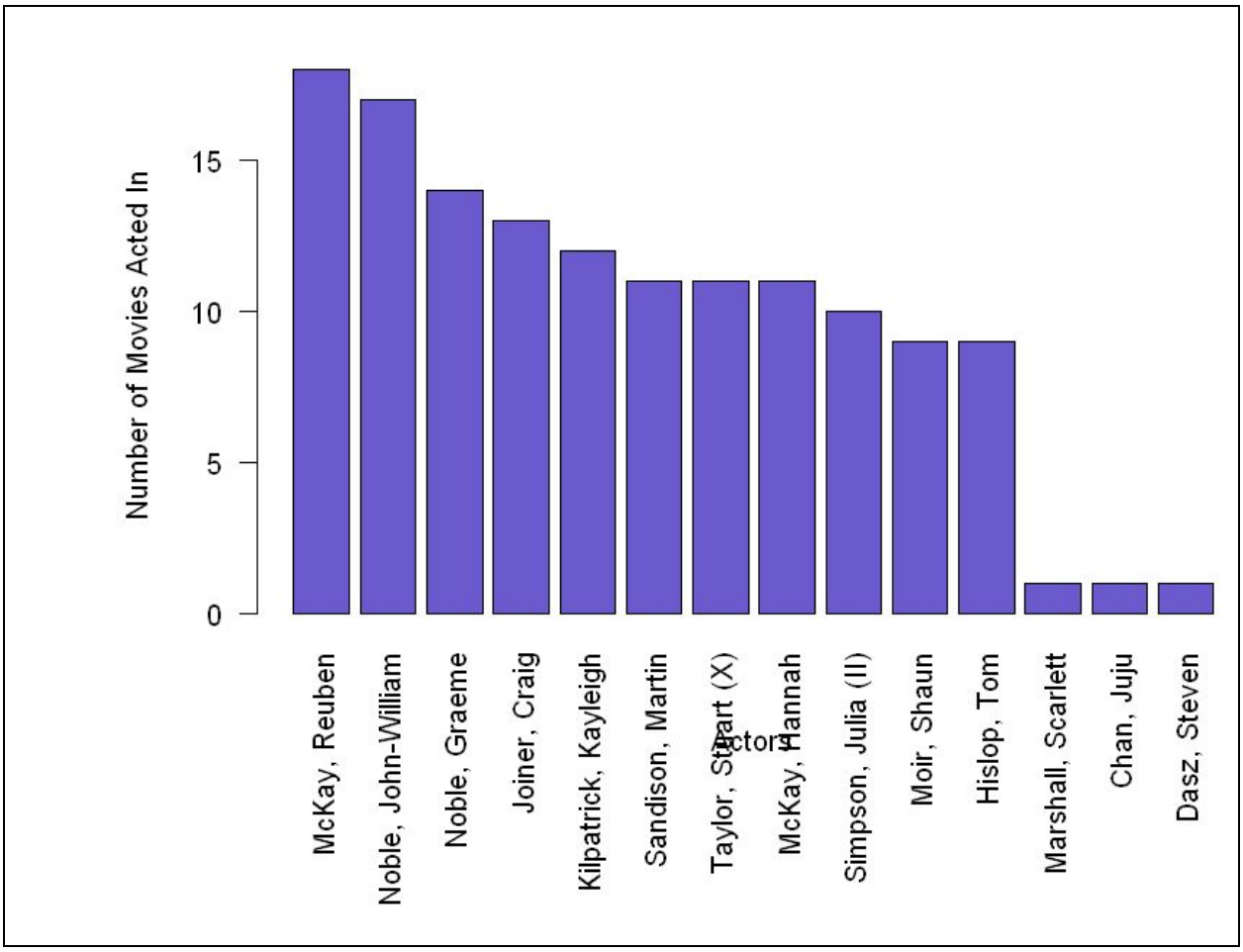

Fear of the Park (2010)
Moir, Shaun
Be My Valentine (2011)
Simpson, Julia (I)
Love House (2011)
Boycie (2011)
The Shadow of Death (2011)
Losers in Love (2011)
Sandison, Martin
The Book of Life (2013/I)
Legion of Evil (2010)
Noble, Graeme
Street Fight (2012)
The Hope Within (2009)
Hislop, Tom
Noble, John-William
Inner Joy of a Broken Heart (2011)
Taylor, Stuart (X)
Call of Babylon (2012)
A Schoolboy Error Production (2009)
McKay, Reuben
McKay, Hannah
Is This It? (2012)
Kilpatrick, Kayleigh
Unconditional Love (2010)
White Cobra (2014)
Booze Culture (2012)
Life of a Spy (2012)

Actor Node



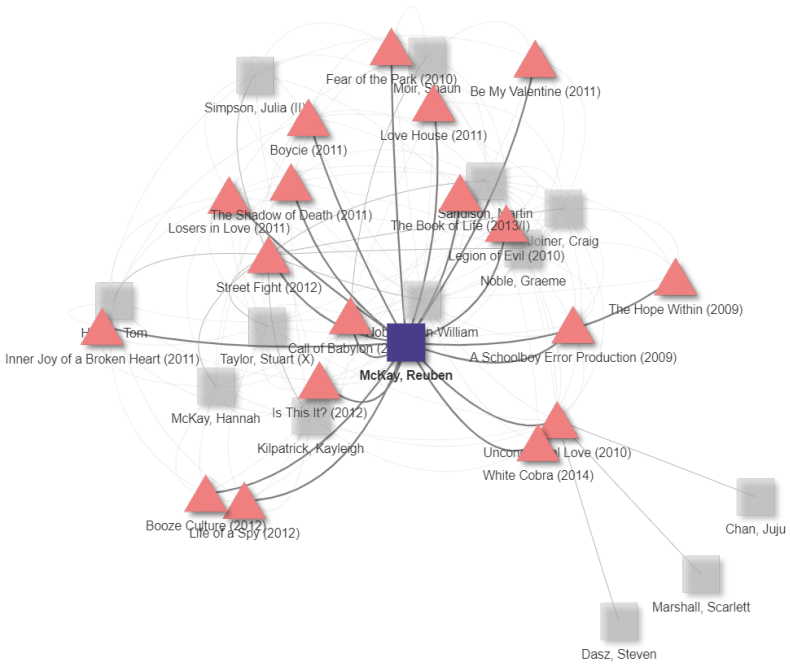
- Next, we find the four most actors in these communities. We do that by inspecting the in degree of all the actor nodes and determining the actors with the maximum in degree as the most important actors for the community.
- The in degrees of the actors is shown below. On the basis of this, the most important actors are:
    - **McKay, Reuben**
    - **Noble, John-William**
    - **Noble, Graeme**
    - **Joiner, Craig**

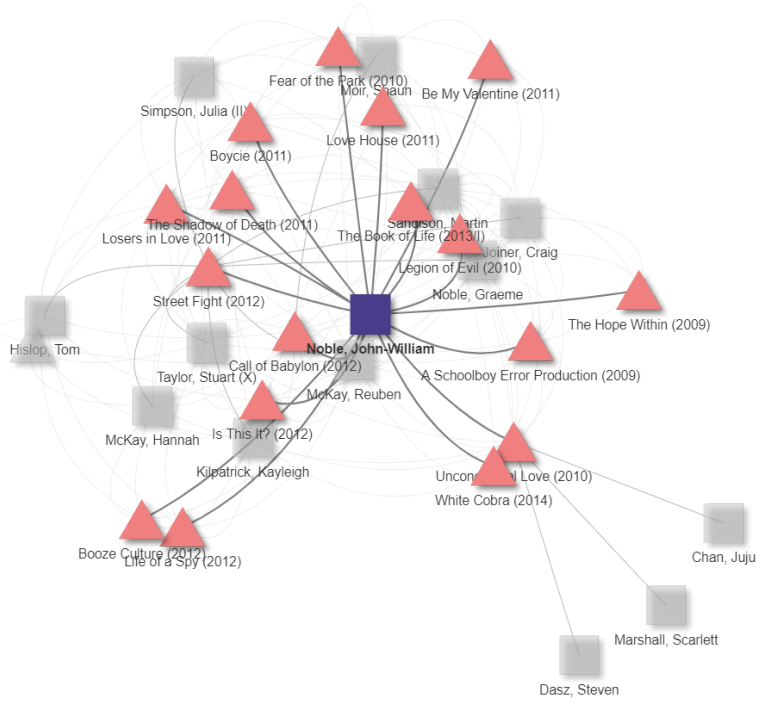
Number of Movies in the Community Associated with the Actors

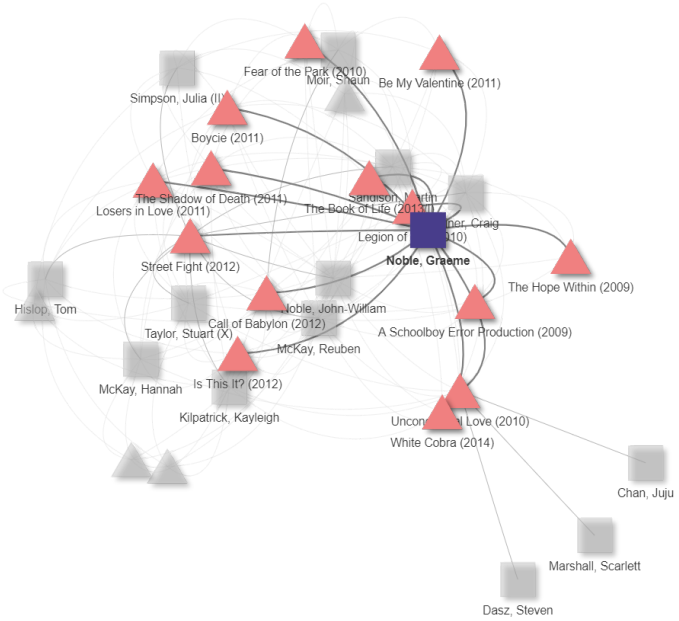


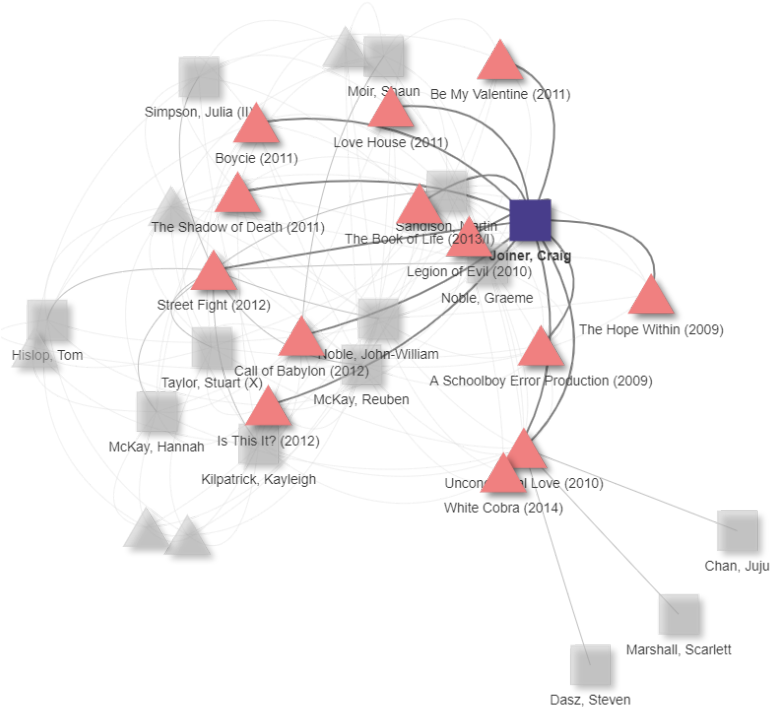
- We further analyse these top actors locally as well as globally (considering all their movies) and obtain the following statistics:

Details of the Top 4 Actors		
McKay, Reuben	**Number of Movies Acted in in this Community**	18

	Visualization of the Movies Acted in in this Community	
	Total Number of Movies Acted in	20
	Dominant Genre Across all Movies	'Comedy': 2, 'Drama': 3, 'Romance': 2, 'Short': 11, 'Thriller': 2
Noble, John-William	**Number of Movies Acted in in this Community**	17

	Visualization of the Movies Acted in in this Community	
	Total Number of Movies Acted in	20
	Dominant Genre Across all Movies	'Comedy': 2, 'Drama': 3, 'Romance': 2, 'Short': 11, 'Thriller': 2
Noble, Graeme	**Number of Movies Acted in in this Community**	14

	Visualization of the Movies Acted in in this Community	
	Total Number of Movies Acted in	18
	Dominant Genre Across all Movies	'Comedy': 2, 'Drama': 3, 'Family': 1, 'Romance': 1, 'Short': 9, 'Thriller': 2
Joiner, Craig	**Number of Movies Acted in in this Community**	13

	Visualization of the Movies Acted in in this Community	
	Total Number of Movies Acted in	13
	Dominant Genre Across all Movies	'Comedy': 1, 'Drama': 3, 'Romance': 1, 'Short': 6, 'Thriller': 2

- We observe that the top actors have acted in predominantly ***Short*** movies in all their movies in the entire movie network. This corresponds to the dominant genre obtained for this community from both question 8(a) and 8(b).
- From these findings, we observe that there is a possible correlation between these actors and the dominant genres obtained, i.e., actors tend to have a particular genre associated with them derived from the genres of movies they act in and when there are movies which feature a lot of actors which have the same genre associated with them, they tend to form communities and the dominant genre for the community will be the dominant genre for all the actors present as can be observed for this community.

  Based on the analysis of both these communities, we can see a clear correlation between the dominant genres of top actors of the community and the dominant genre of the community.

2.3. Neighborhood analysis of movies

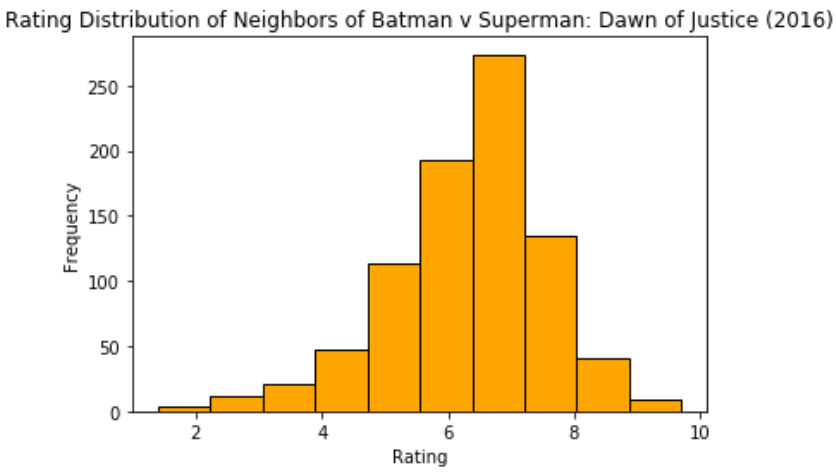
In this part of the project, you will need to load the movie_rating.txt file and we will explore the neighborhood of the following 3 movies:

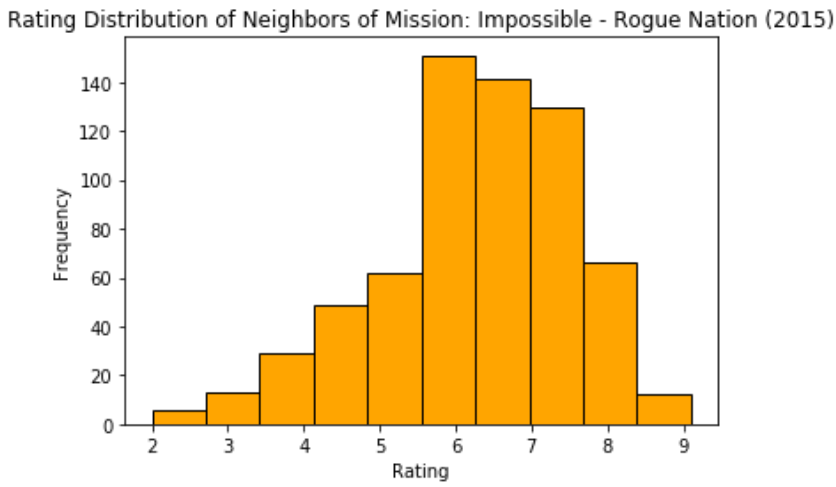
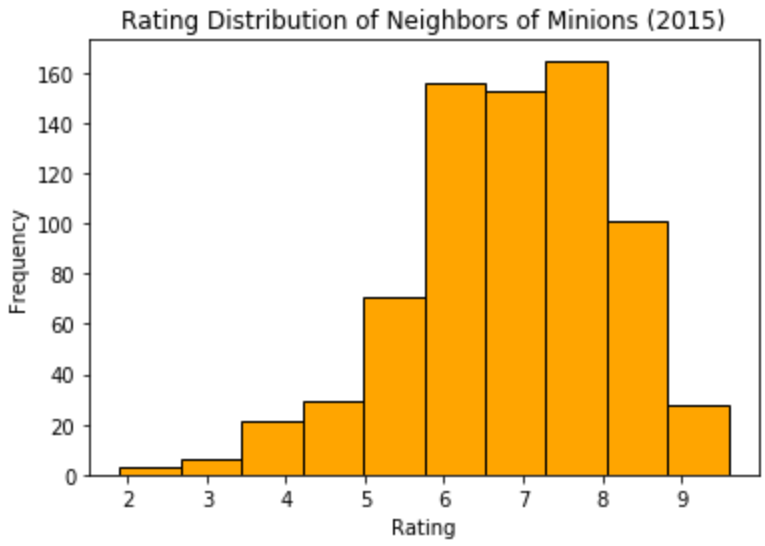
- Batman v Superman: Dawn of Justice (2016); Rating: 6.6
- Mission: Impossible - Rogue Nation (2015); Rating: 7.4
- Minions (2015); Rating: 6.4

9) For each of the movies listed above, extract it's neighbors and plot the distribution of the available ratings of the movies in the neighborhood. Is the average rating of the movies in the neighborhood similar to the rating of the movie whose neighbors have been extracted? In this question, you should have 3 plots.

Ans:

In this problem, we have extracted the neighbors of the movies and calculated the average rating of the neighbors. The distribution of ratings and the average have been documented in the table below:

Batman v Superman: Dawn of Justice (2016)	***Number of Neighbors***	1051
	Neighbors Having Rating Information	850
	Average Rating	6.328705882352948
	Distribution of Ratings	
Mission: Impossible - Rogue Nation (2015)	***Number of Neighbors***	803
	Neighbors Having Rating Information	659

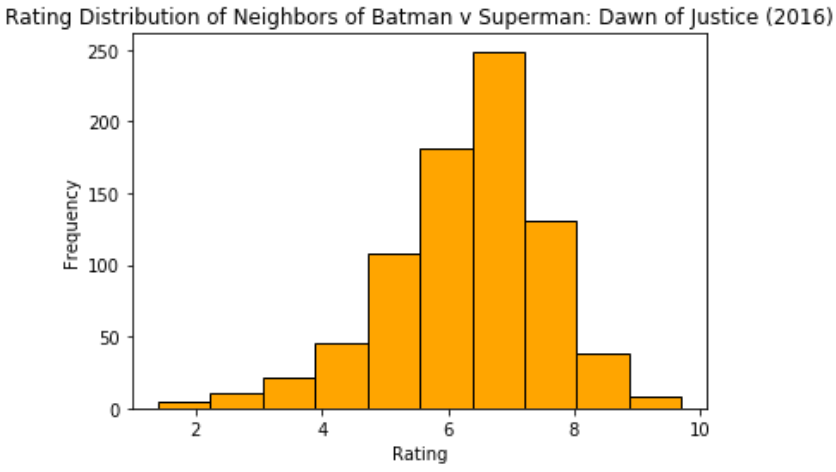
	Average Rating	6.236874051593324
	Distribution of Ratings	
Minions (2015)	***Number of Neighbors***	772
	Neighbors Having Rating Information	733
	Average Rating	6.829331514324697
	Distribution of Ratings	

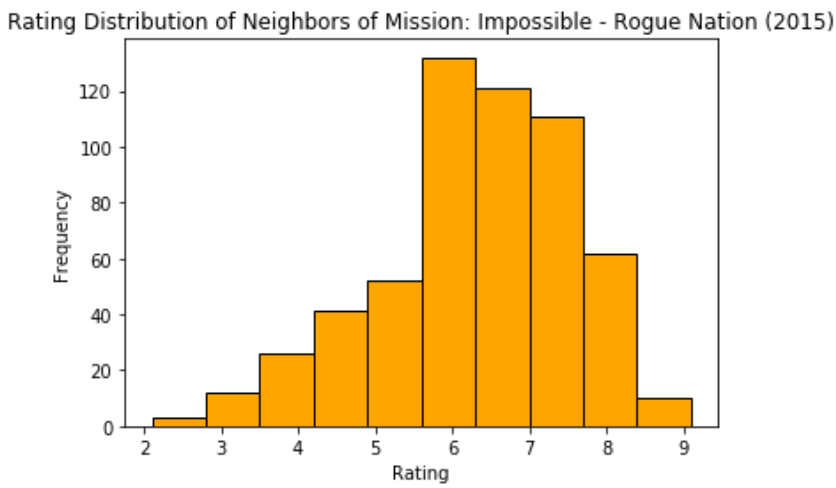
We observe that the average rating of the neighbors of each of these movies are significantly different from the actual rating of the movies in question. This indicates that the average rating of the neighbors is not a good indicator in determining or predicting the rating of a movie.

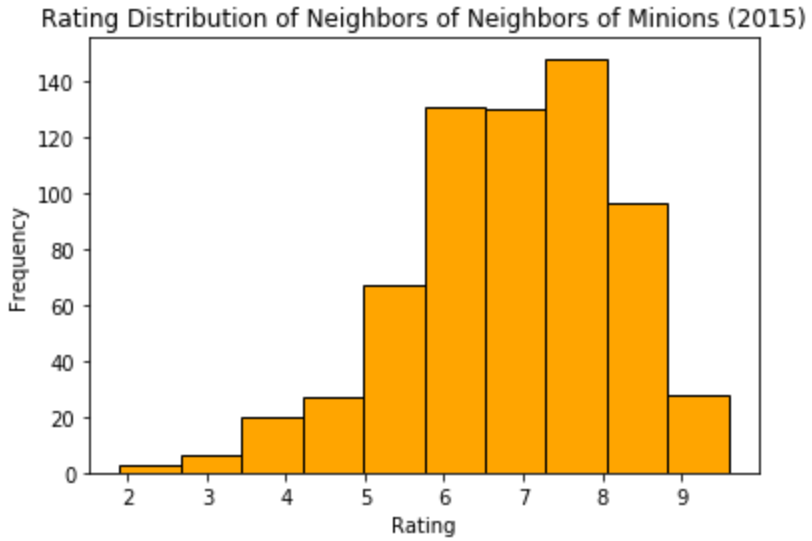
10) Repeat question 10, but now restrict the neighborhood to consist of movies from the same community. Is there a better match between the average rating of the movies in the restricted neighborhood and the rating of the movie whose neighbors have been extracted. In this question, you should have 3 plots.

Ans

We observe that all the three movies belong to the same community. This community also happens to be the largest community of all. In this problem we further filter the neighbors of the movie by considering neighbors which are in the same community as the movie in question. The results are shown below:

Batman v Superman: Dawn of Justice (2016)	***Community It Belongs To***	2
	Number of Neighbors	1051
	Neighbors Having Rating Information	850
	Neighbors in the Same Community	985
	Neighbors in the Same Community with Rating Information	796
	Average Rating	6.319472361809057
	Distribution of Ratings	
Mission: Impossible - Rogue Nation	***Community It Belongs To***	2
	Number of Neighbors	803

(2015)	***Neighbors Having Rating Information***	659
	Neighbors in the Same Community	706
	Neighbors in the Same Community with Rating Information	570
	Average Rating	6.263333333333333
	Distribution of Ratings	
Minions (2015)	***Community It Belongs To***	2
	Number of Neighbors	772
	Neighbors Having Rating Information	733
	Neighbors in the Same Community	694
	Neighbors in the Same Community with Rating Information	656
	Average Rating	6.8408536585365916

	Distribution of Ratings	

We do not observe any significant variation in the results with respect to the results obtained in the previous problem. This is due to the reason that even on filtering, the number of neighbors to consider is still large, i.e. most of the neighbors are in the same community as the movie in question. Due to this we don't see a significant change in the average rating again indicating that this could be a poor indicator for prediction of movie ratings.

11) For each of the movies listed above, extract it's top 5 neighbors and also report the community membership of the top 5 neighbors. In this question, the sorting is done based on the edge weights.

Ans:

The details of the top 5 neighbors for each of the movies is given below:

Batman v Superman: Dawn of Justice (2016) - Community Number 2					
Movie ID	***Movie Name***	***Edge Weight***	***Movie Community***	***Rating***	***Average Rating***
26906	Eloise (2015)	0.1125	2	4.5*	6.24
12646	The Justice League Part One (2017)	0.075758	2	6.7*	
40754	Into the Storm (2014)	0.07292	2	7.1	
11626	Love and Honor (2013)	0.060976	2	5.7	
4334	Man of Steel (2013)	0.059829	2	7.2	

Mission: Impossible - Rogue Nation (2015) - Community Number 2					
Movie ID	***Movie Name***	***Edge Weight***	***Movie Community***	***Rating***	***Average Rating***
40106	Fan (2015)	0.1585366	2	7.2*	6.98
40107	Phantom (2015)	0.1460674	2	5.7*	
72245	Breaking the Bank (2014)	0.1028037	2	8.6	
87079	Suffragette (2015)	0.1022727	2	6.9	
48390	Now You See Me: The Second Act (2016)	0.1011236	2	6.5*	
Minions (2015) - Community Number 2					
Movie ID	***Movie Name***	***Edge Weight***	***Movie Community***	***Rating***	***Average Rating***
46198	The Lorax (2012)	0.25	2	6.5	7.48
20468	Inside Out (2015)	0.25	2	8.9	
65505	Up (2009)	0.2368421	2	8	
77039	Surf's Up (2007)	0.2368421	2	6.5	
46219	Despicable Me 2 (2013)	0.225	2	7.5	

* The movie rating was taking from imdb.com as it was not available in the rating file.

- Some key observations from the results obtained are
    - The top 5 neighbors for each community belong to the same community as the movie itself. This indicates that edge weight plays a significant role in community formation.
    - Additionally, we can intuitively see why the movies selected are the top neighbors - for Inside Out, we expect other animated movies to be its top neighbors which is confirmed in the results. Similarly, for Dawn of Justice we see the occurrence of two DC related movies and Mission Impossible being an action movie, its top neighbors are also action movies even if they are not of the same language
    - Again, the average ratings of the neighbors are not a good feature for predicting movie ratings. The results in this question and as well as the previous two questions prompt us to explore different features for developing a system for prediction of movie ratings which we further do in the next two questions.

2.4. Neighborhood analysis of movies

In this part of the project, we will explore various rating prediction techniques to predict the ratings of the following 3 movies:

- Batman v Superman: Dawn of Justice (2016)
- Mission: Impossible - Rogue Nation (2015)
- Minions (2015)

12) Train a regression model to predict the ratings of movies: for the training set you can pick any subset of movies with available ratings as the target variables; you have to specify the exact feature set that you use to train the regression model and report the root mean squared error (RMSE). Now use this trained model to predict the ratings of the 3 movies listed above (which obviously should not be included in your training data).

Ans:

In this task, we have used only a subset of the movies from the movie_rating dataset. We have used some features which require actor information as well, so we have taken only the subset of the movies from movie_rating for which we have actor information as well. Of this subset we have taken 10,000 movies for our regression task.

We have used the following features for regression.

1. **The year of the movie** - There might be some relation between the ratings and how new/old the movie is.
2. **Number of actors in the movie** - A movie with many actors and cast might affect the ratings of the movie
3. **The Actors in the movie** - The actual actors who have worked in the movie. This is basically represented as a one-hot encoded vector where 1 denotes that the actor at the particular index is present in the movie, 0 otherwise.
   So this is a large vector of binary values of length 113,110 x 1
4. **Movie genre** - We have used the movie genre information by label encoding the movie genres. There are total 29 genres we have extracted from the movie_genre file and we use a unique numerical id assigned to each genre as a feature.

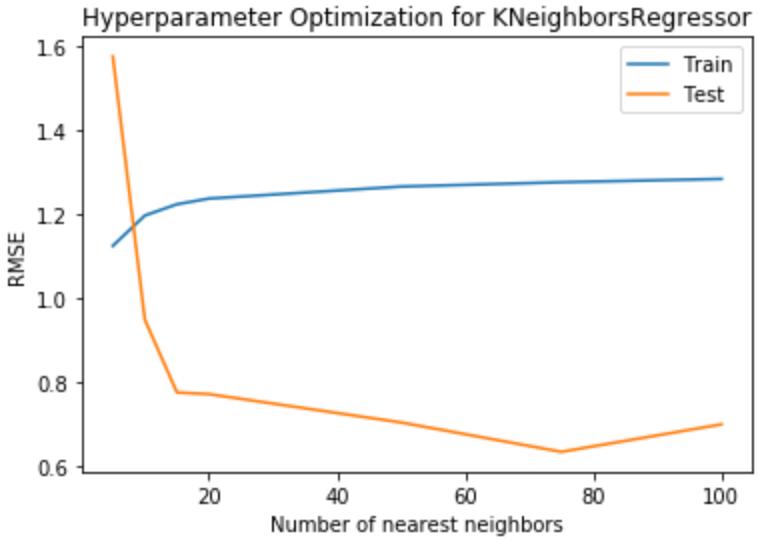
As the scale of the features vary, we have used **Standard Scalar** which normalizes the features. Standardization of features is done by removing the mean and scaling to unit variance. Centering and scaling is done independently on each feature by computing the relevant statistics on the samples in the training set. Standardization of a dataset is a common requirement for many machine learning estimators: they might behave badly if the individual feature do not more or less look like standard normally distributed data (e.g. Gaussian with 0 mean and unit variance).

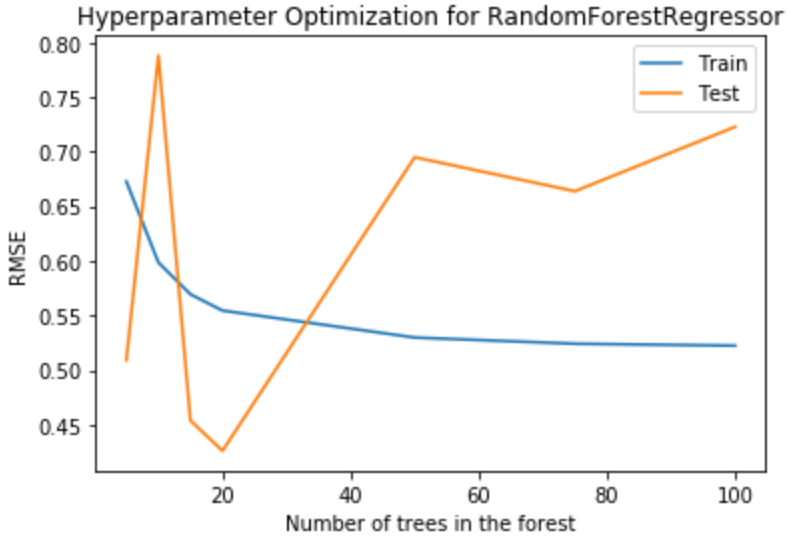
As we have a large vector of features, especially due to the actor vector, we have used **Principal Component Analysis(PCA)** to find the principal components. PCA uses linear dimensionality reduction using Singular Value Decomposition of the data to project it to a lower dimensional space. Using this we have reduced the dimensionality of our feature vector from **113,113 to 200**.

We have experimented with different Regression models as well.

1. **Linear Regression:** Ordinary least squares Linear Regression.
2. **KNeighborsRegressor:** Regression based on k-nearest neighbors. The target is predicted by local interpolation of the targets associated of the nearest neighbors in the training set.
3. **RandomForestRegressor:** A random forest is a meta estimator that fits a number of classifying decision trees on various sub-samples of the dataset and use averaging to improve the predictive accuracy and control over-fitting.

Once the models were chosen, we have tuned the hyperparameters of the models to get the best train RMSE.





The results from the different models is presented in the tables below. We also predict the ratings for the 3 movies using the different regression models.

Model	**Train RMSE**	**Test RMSE**
Linear Regression	1.3228	0.7710
Rating Predictions		
Batman v Superman: Dawn of Justice (2016)	**Mission: Impossible - Rogue Nation (2015)**	**Minions (2015)**
6.1591	6.16254497	6.1595

Model	Train RMSE	Test RMSE
KNeighborsRegressor	1.1233	1.5735
Rating Predictions		
Batman v Superman: Dawn of Justice (2016)	**Mission: Impossible - Rogue Nation (2015)**	**Minions (2015)**
4.6	5.74	5.58

Model	Train RMSE	Test RMSE
RandomForestRegressor	0.5223	0.7227
Rating Predictions		
Batman v Superman: Dawn of Justice (2016)	**Mission: Impossible - Rogue Nation (2015)**	**Minions (2015)**
6.438	6.224	6.002

13) Create a bipartite graph following the procedure described above. Determine and justify a metric for assigning a weight to each actor. Then, predict the ratings of the 3 movies using the weights of the actors in the bipartite graph. Report the RMSE. Is this rating mechanism better than the one in question 12? Justify your answer.

Ans:

Even in this part, we have taken the same subset of data(10,000 movies) that we have used in the previous part for accurate comparison.

Instead of explicitly constructing a bipartite graph, we have used the underlying concept behind it to get the representation of the graph. The bipartite graph ensures that there are 2 sets - actors and movies, with no edges within the set i.e. among the actors and among the movie set themselves but the edges are present from the actors to the movies and vice versa. We have a mapping of the movies that an actor has worked in. These are basically the edges from the actor vertex to the

respective movie vertex in the bipartite graph. Similarly, we have a mapping from movies to all the actors that have worked in that movie. These are the edges from the movie vertices to the actor vertices. So with these two mappings we basically have the adjacency list representation of the bipartite graph.

Let us consider that the rating of the movie is basically the weight of the respective movie vertex. Consider the following facts for assigning weights to the actor vertices

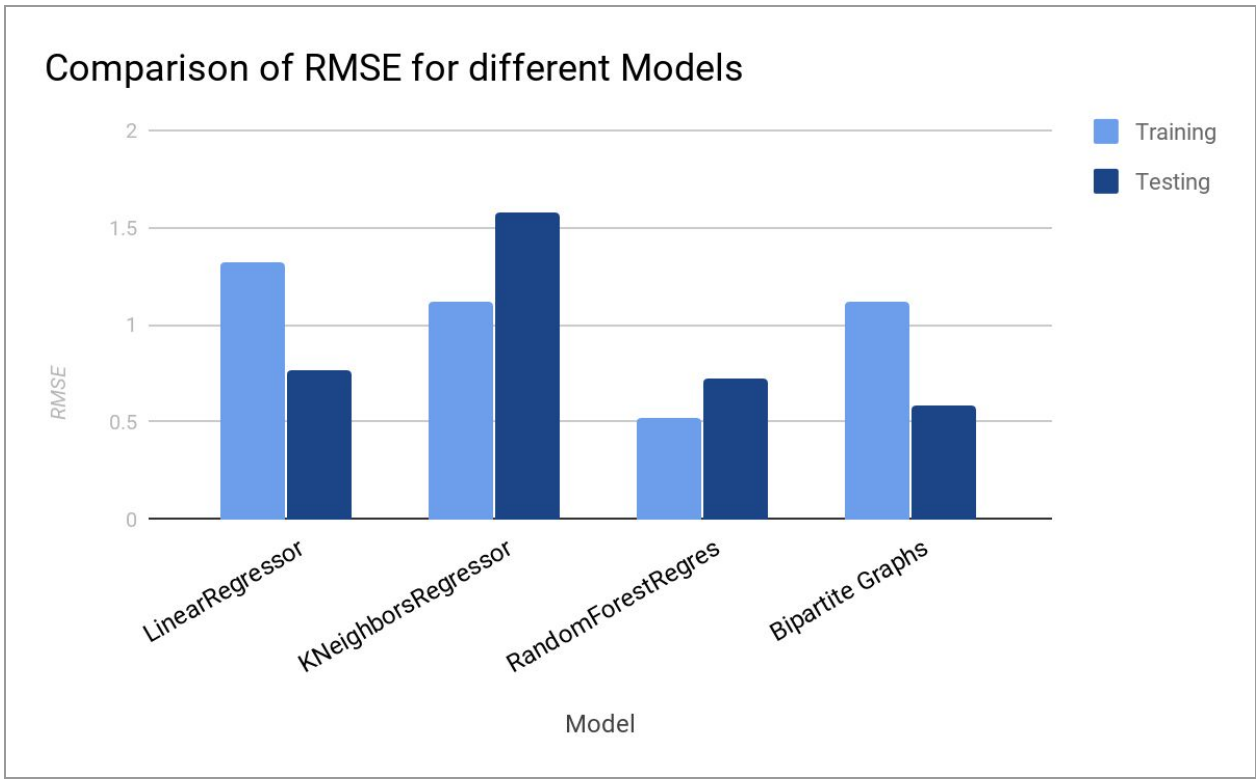
1) Now as observed in real life, the actors who are present in the movie have a great influence on the rating of a movie.
2) The quality of an actor can be judged from the movies he/she does and the ratings and popularities of these movies.

Using these 2 facts, we assign an actor a weight equal to the average of the movie ratings of the movies that the actor has worked in. This feels intuitive as well as deducible from the available data. If an actor has worked in many movies with high ratings, then the actor must have had some contribution which makes all the ratings high and thus, we assign a high weight to this actor. On the other hand, if an actor has by coincidence been a part of a high rating movie, but all other movies by the actor has low ratings, then as we take average of all the ratings, the actor will automatically be assigned a low weight.

Once we have assigned such weights to the actors, we need to predict the ratings for the new movies. Using the first fact, we predict ratings for new movies by taking the average of actor weights of the actors that have worked in this movie. Using this method of rating movies, we have predicted the ratings for the training set as well as the 3 movies in the test set. The details are presented in the table below.

Method	**Train RMSE**	**Test RMSE**
Bipartite Graphs	1.1172	0.5853
Rating Predictions		
Batman v Superman: Dawn of Justice (2016)	**Mission: Impossible - Rogue Nation (2015)**	**Minions (2015)**
6.496	6.499	6.855

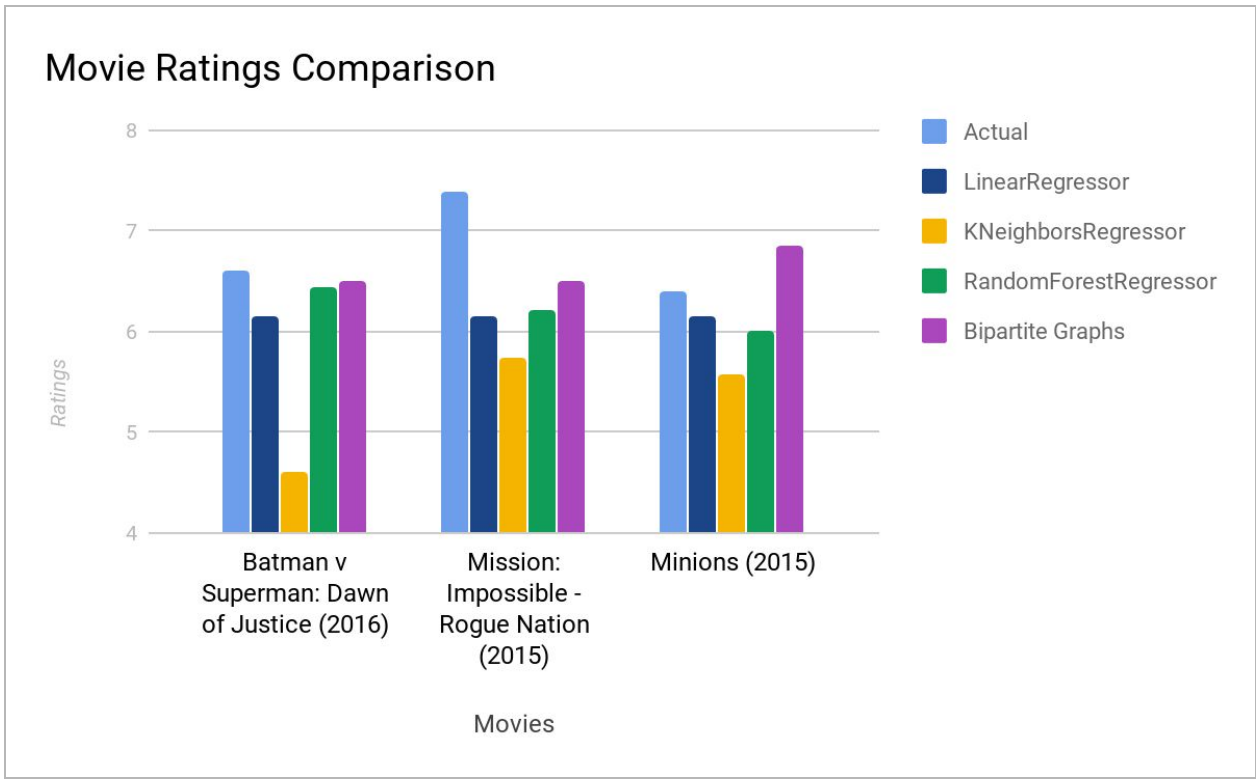
We will now compare the different methods and models for movie rating prediction.



Looking at the RMSE results, we can say that the method of bipartite graphs is equally good as compared to normal regression methods. In normal regression we have used the actor information as features while in the bipartite graph as well, we assign the weights to the actors based on the ratings of the movie. Thus, there is a similar concept in both the methods and so we get similar results.

However the random forest regressor works gives a lower RMSE than the remaining three and gives better results. The training RMSE for random forests is very low as compared to the others because this model usually overfits the training data. The testing RMSE of random forests is good as well and we get good predictions for the movies. This works better than the bipartite graphs method as apart from just the actor ratings, we use also include some additional features regarding the year, genre, presence/absence of individual actors etc.

The graph below shows the comparison between the ratings predicted by the different models.


Movie Ratings Comparison
8
7
6
5
4
Ratings
Batman v Superman: Dawn of Justice (2016)
Mission: Impossible - Rogue Nation (2015)
Minions (2015)
Movies
Actual
LinearRegressor
KNeighborsRegressor
RandomForestRegressor
Bipartite Graphs