Recommending Movies Using Hybrid Approach

Mounik Patel
Lakehead University - Computer Science
Email : mpatel78@lakeheadu.ca
Student ID : 1144127

Abstract— **With the rapid explosion of video streaming platforms on the Internet, the catalog of movies is rising exponentially, leaving viewers overwhelmed with a huge database of movies to choose from. Movie Recommendation Systems come into play, which consider users' preferences and recommend movies to them. This saves users a lot of time and effort that would otherwise be wasted while searching for a movie manually. The research paper defines various terms and approaches to recommendation systems. Various recommendation algorithms used by Netflix are reviewed, which makes the video streaming platform more efficient and popular. A brief description of a hybrid approach to movie recommendation systems that is implemented by combining content-based and collaborative filtering approaches is given, where content-based movie recommender systems consider movie information and collaborative filtering movie recommender systems consider user ratings for movie recommendations. Four different implementations of movie recommender systems were explained, evaluated using a user survey, and their results were compared and discussed.**

Keywords – Movies, Recommendation Systems, Content-Based, Collaborative Filtering, Hybrid

I. INTRODUCTION

A Recommendation System is a software tool or technique that assists users in finding the items they want from the large number of items available on the Internet [1]. The users are given personalized recommendations as a ranked list of items in their simplest form. The recommendation systems try to predict the most suitable product or service, according to users' choices and constraints. To carry out this computational task, recommendation systems gather users' preferences that are either expressed directly (product ratings) or inferred by user actions (user browsing through specific products) [2]. Companies like Netflix, Spotify, Amazon, etc., utilize the recommendation system to boost customers' satisfaction and interactivity, which helps these businesses achieve their financial goals [3]. Building an efficient recommendation system is a challenging task because the user's preferences keep changing over time.

Recommendation systems need to gather various types of data to give precise and better recommendations. The data used for the recommendations concerns primarily the items and the users to which these recommendations are provided. Recommendation systems utilize three types of objects while collecting or processing data: items, users, and transactions, i.e., user-to-item relationships [2].

The most general context for the study of recommendation systems is shown in the Fig. 1. Known user preferences are represented as a matrix of n users and m items in which each cell $r_{u,i}$ corresponds to the rating given to item i by the user u. This matrix is known as the user ratings matrix, which is usually sparse since most users do not evaluate most of the items. The recommendation task is to predict what rating a user would give to a previously unrated item. Typically, ratings of all items not seen by a user are predicted, and recommendations are provided for the highest rated items. The user under current consideration for recommendations is referred to as the active user [4].

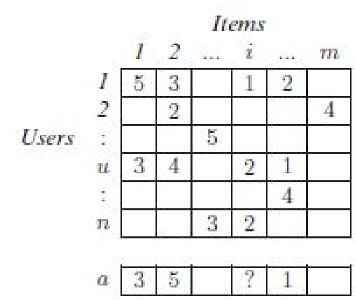

Fig. 1. User ratings matrix [4], where each cell $r_{u,i}$ corresponds to the rating of user u for item i. The task is to predict the missing rating $r_{a,i}$ for the active user a.

In analyzing various data sources, recommendation systems differ to establish ideas of affinity between the users and items that can be used to identify well-matched pairs. This leads to various recommendation systems approaches that can be widely classified as [4]:

1) Content-based: The system in this approach can recommend items similar to the ones that the user liked in the past. The similarity of the items is calculated based on the features of the compared items. For example, if a user has positively rated a movie that belongs to the comedy genre, then the system can learn to recommend other movies from this genre [2].
2) Collaborative filtering: This approach recommends items to the active user that other people have liked in the past. The similarity between the two users' preferences is calculated based on the similarity in the users' rating history [5].
3) Hybrid approaches: These recommendation systems are based on a combination of the above-mentioned techniques. A hybrid system combining the techniques of model A and model B aims to use the benefits of

model A to fix model B's drawbacks. For example, collaborative filtering approaches suffer from new-item problems, i.e., they do not recommend items without ratings. Content-based approaches do not have this limitation, since the prediction of new items is based on their descriptions that are easily available [2].

The research primarily focuses on movie recommendation systems. Movies can be easily differentiated through their genres, such as comedy, romance, thriller, action, etc. Another way for movies to be categorized is by their metadata, such as release year, language, director, or cast [6]. Movie recommendation systems help users to search for preferred movies among all the movies available, and hence save users from the trouble of wasting a lot of time searching for preferred movies. This necessitates that the movie recommendation system be very reliable and offer recommendations for movies that are exactly the same or most matched with users' preferences. Building a movie recommendation system is challenging since there are a lot of movies and a large number of users watch them [7]. A movie recommendation system should therefore provide a level of comfort and personalization that will assist the user to better interact with the system and recommend movies that meet his needs [8].

II. BACKGROUND

This section reviews literature related to The Netflix Recommender System, which describes the various movie recommendation algorithms used by Netflix, a popular movie streaming platform.

A. The Netflix Recommender System: Algorithms, Business Value, and Innovation

This research paper detailed the various algorithms that make up the recommendation system of Netflix and outlined its purpose for business. The problems faced by recommendation systems have been studied and an approach utilized to improve the recommendation algorithms has been further reviewed using the A/B testing method. Recommendation systems are an important part of Netflix, as they enable customers to find content to watch in every session. The essence of the whole Netflix experience is due to a collection of different algorithms serving different use cases [9].

With the advent of Internet TV, users are getting a lot of choices in terms of content-to-watch. But this plethora of choices has also led users to waste their time by browsing through various movies before finalizing a specific movie to watch. Netflix realized this recommendation problem in the past and came up with a similar problem, where the Netflix recommendation problem predicts the number of stars that a person will rate a video on a scale from 1 to 5. A competition (Netflix Prize 2009) was also organized to improve the accuracy of the rating prediction. That resulted in various algorithms for recommendations which are still used till date. Recommendations provided with the help of ratings for content have been outdated since the outburst of the vast amount of data generated daily which describes various details about the user, their watching-activity, and the content they watched and rated. These vast amounts of data led to the improvement of Netflix as a service that helps users find more videos to watch rather than only recommending videos with a high star rating [9].

The recommendation system of Netflix comprises of a variety of algorithms that collectively determine the Netflix experience, which can be seen on the Netflix homepage. It has a matrix-like layout, where each entry in the matrix is a recommended video, and each row of videos contains recommendations with a similar "theme". Typically, the videos in a single row come from a single algorithm [9]. Some of the recommendation algorithms utilized by Netflix are listed in this research paper [9]:

- Personalized Video Ranker (PVR): This algorithm, as its name implies, personalizes the whole video inventory for each user profile. The resulting ordering is used to select the orders of the videos in genres and other rows. This is the reason different users have completely different videos in the same genre row.
- Top-N Video Ranker: This algorithm generates the recommendations in the Top Picks row. The purpose of this algorithm is to identify the best few personalized recommendations in the entire catalog for each member, i.e., focused only on the head of the ranking. The Top-N ranker is optimized and assessed using metrics and algorithms which only check the head of the catalog ranking provided by the algorithm and not the complete catalog ranking (in the case of PVR).
- Trending Now: Shorter-term temporal trends ranging from just a few minutes to a few days are great video predictors, particularly if combined with a suitable level of personalization, providing the user with a trend ranking. The kinds of trends these rankers identify can vary, with a few trending every several months, but most of them trending just once in a while.
- Continue Watching: The Continue Watching ranker sorts the subset of recently viewed titles based on the best estimate of whether the user is going to resume watching or rewatching, or if the user has given up something not as interesting as expected.
- Video-Video Similarity: A Because You Watched (BYW) row anchors its recommendations to a single user-watched video. The video-video similarity algorithm (referred to as "sims") provides the recommendations in these rows. The Sims algorithm is an unpersonalized algorithm that computes a ranked list of similar videos for every video in the catalog.

All the above algorithms were tested with the A/B testing method, where users (existing users and new users) received two variants (A and B) of the Netflix homepage for browsing and watching videos. The test has been done for a month, following which the test results show how much a user has interacted and satisfied after browsing through the whole video catalog and checking out the recommended content from Netflix [9].

Researchers concluded that recommendation systems will continue to play a key role in utilizing the vast amount of data available to make these choices manageable and successfully guide users towards the best possible options to evaluate, which results in better decisions [9].

III. METHODOLOGY

A. Content-Based Approach

Content-based recommendation systems analyze movie descriptions in order to recommend movies that are of particular interest to the user. A content-oriented approach has the advantage of not having other users' information or data, and the recommender system will recommend new movies or movies that the user has not yet watched [10]. This approach works best when there is known information about a movie (title, genre, release year, etc.) but not about the user. Content-based recommenders handle recommendations as a user-specific classification problem, learning a classifier for the user's likes and dislikes based on the features of a movie. This method is based on research in information retrieval and information filtering [11].

B. Collaborative Filtering Approach

The collaborative filtering system recommends movies based on similarity measures between users and/or movies. The collaborative filtering method tries to find other like-minded users and then recommends the movies that are most liked by them. This is based on a scenario where a person asks his friends, who have similar tastes, to recommend him some movies [12]. While the term ”collaborative filtering” has only been around for a little more than a decade, it derives from something humans have been doing for hundreds of years: sharing opinions with others. Collaborative filtering predicts user preferences in movie recommendations based on previously known user-ratings of movies [10]. A key advantage of the collaborative filtering approach is that it does not rely on machine analyzable content and, as a result, can accurately recommend complex items like movies without requiring an ”understanding” of the item itself. Many of the algorithms have been utilized in collaborative filtering recommender systems to measure user or movie similarity [11]. Two of the similarity measures used in this research are as follows [13]:

1) Cosine Similarity: The measure of similarity between two non-zero vectors in the inner product space is cosine similarity. It measures the angle formed by these two vectors. The cosine of two non-zero vectors can be calculated using their dot products:

$$u.v = ||u||\ .\ ||v||\ .\ cos\theta$$

Cosine similarity is particularly beneficial in positive space, where the result is efficiently constrained in the range [0, 1]. Thus, for two given vectors u and v, the cosine similarity, $cos\theta$ can be calculated as the product of the vectors' dot product and magnitude:

$$sim(u,v)^{cosine} = cos\theta = \frac{\sum u\ v}{||u||\ .\ ||v||}$$

2) Pearson Correlation: It determines user similarity using the Pearson Correlation Coefficient. The higher the coefficient, the more closely related the two users are. The Pearson Correlation Coefficient formula is given below:

$$r = \frac{\sum((u-\bar{u})(v-\bar{v}))}{\sqrt{\sum(u-\bar{u})^2 . \sum(v-\bar{v})^2}}$$

C. Hybrid Approach

To minimize constraints and enhance the significance of movie recommendations, this research analyzes and combines both content-based and collaborative filtering approaches. The hybrid approach's basic model is depicted in Fig. 2

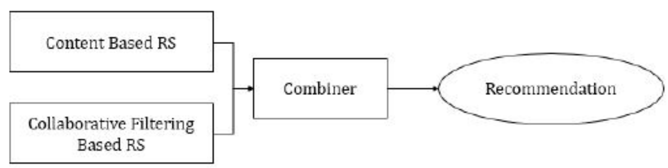

Fig. 2. Hybrid Movie Recommendation System combining Content-Based and Collaborative Filtering Approaches

IV. IMPLEMENTATION

A. Dataset Description

Two different datasets from MovieLens that were collected by the GroupLens Research team for research work in the field of recommender systems were used. They are as follows [14]:

1) MovieLens 25M Dataset: Approximately 63 thousand movies and 25 million user-ratings
2) MovieLens Latest Small Dataset: Approximately 10 thousand movies and 100 thousand user-ratings

The MovieLens Dataset involves the following:

- Movie ratings by each user are assigned from 0.5 to 5 (half-increment).
- Each user has rated at least 20 movies.
- All users were selected at random, and there is no demographic information included. Each user has been labeled with a unique userId, and no other information is revealed.

The MovieLens dataset was chosen primarily because it is publicly available and has been used in many hybrid recommender systems, making it a good baseline for this purpose.

B. Data Preprocessing

The data was preprocessed before being used in the research project's implementation. Unnecessary columns were removed in order to reduce computational expenses. There was no release year column because the year was included in the movie title. As a result, the release year was extracted from the movie title and placed separately in the year column. To improve data processing, all null values were replaced with 0. The datatypes for movie years and movie genres were changed to integer and string, respectively.

C. Implementation of Content-Based Movie Recommender Systems

The top-N popular and top-N rated movies form the basis for the content-based movie recommendation system. This recommendation system recommends movies based on the user's inputs, such as the number of movies to be recommended, the year range, and the genre. Movies will be chosen from the dataset based on the genre and year range provided as input. For each of the movie recommender systems, the selected movies are filtered using the following criteria:

1) Top-N Popular Movies Recommender System: The Top-N Popular Movies Recommender System considers the average rating of each movie as well as the total number of users who have previously rated the movie. Movies chosen by users based on year range and genre will be further filtered based on average ratings for all movies and the average number of users who rated each movie. Finally, the filtered movie list will be sorted in descending order based on the number of users that rated each movie. This will return the top-N most popular movies as recommendations, where popularity is defined as the number of users who have rated each movie.
2) Top-N Rated Movies Recommender System: The Top-N Rated Movies Recommender System takes into account each movie's average rating as well as the total number of users who have previously rated the movie. Movies chosen by users based on year range and genre will be further filtered based on overall average ratings and the average number of users who rated each movie. Finally, based on the average rating provided by users, the filtered movie list will be ordered in descending order. This will return the top-N most rated movies as recommendations, where rating is defined as the average rating of each movie.

D. Implementation of Collaborative Filtering Movie Recommender Systems

The top-N similar movies serve as a basis for the collaborative filtering movie recommendation system, with similarity defined as ratings given to other movies by users who have also rated a given movie. This recommendation system recommends movies depending on the user's inputs, which include the number of movies to be recommended, the year range, and the movie title with release year. Movies will be filtered based on the year range provided as input as well as similarity metrics, which will aid in the identification of similar movies based on users who have already rated the given movie. The selected movies are filtered using the following criteria for each of the movie recommender systems:

1) Top-N Similar Movies Recommender System (by Cosine Similarity): The Top-N Similar Movies Recommender System (by Cosine Similarity) takes into account the cosine similarity score between the user's rating of the input movie and the ratings of movies for which other users have also rated the input movie. Movies chosen by users based on year range will be translated into a user-ratings matrix, which will detail each user's rating of a movie. User ratings for the input movie will be filtered, and other users who have rated the input movie will be selected. Movies rated by these selected users will be given further consideration for recommendations. The recommended movies will be further filtered by picking only the top 0.01% of the total number of users that rated each movie, and they will be sorted on the basis of cosine similarity score.
2) Top-N Similar Movies Recommender System (by Pearson Correlation): The Top-N Similar Movies Recommender System (by Pearson Correlation) considers the Pearson correlation coefficient between the user's rating of the input movie and the ratings of movies that other users have also rated. Movies selected by users based on year range will be translated into a user-ratings matrix, which will describe each user's rating of a movie. Other users who have rated the input movie will be chosen after filtering user ratings for the input movie. Movies rated by these users will be given further consideration for recommendations. The recommended movies will be further filtered by selecting only the top 0.01% of all users who rated each movie, and they will be ranked based on Pearson correlation score.

V. RESULTS

Human evaluation was deemed as an evaluation metric because it was an effective way to directly understand the user's preferences in terms of how well a movie recommender is performing. Ten users were chosen at random to evaluate all four recommenders used in this research. The number of movie recommendations was fixed at 10 for all users to ensure consistency when comparing results.

A. Evaluation of Content-Based Movie Recommender Systems

Other inputs (year-range and genre) were also kept fixed during the evaluation of both content-based movie recommender systems, namely the Top-N Popular Movies Recommender System and the Top-N Rated Movies Recommender System. This would provide consistency when comparing the results of both content-based movie recommender systems. The results of the users' survey for the Top-N Popular

Movies Recommender System and Top-N Rated Movies Recommender System are shown as a bar-plot in Fig. 3 and Fig. 4, respectively.

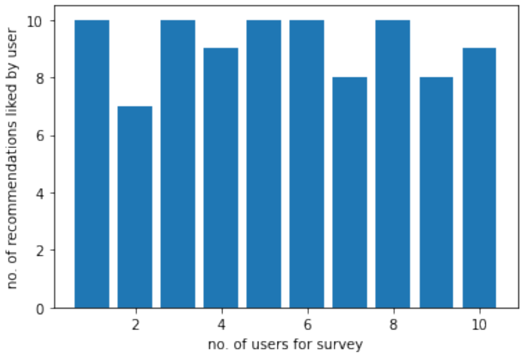

Fig. 3. User-Survey: Top-N Popular Movies Recommender System

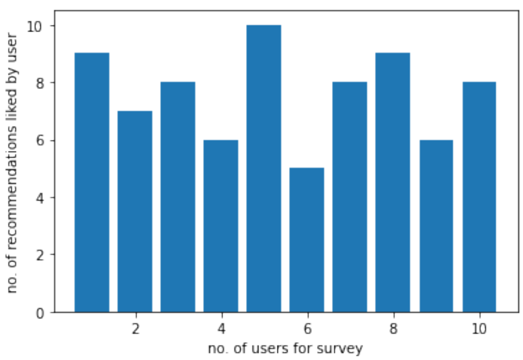

Fig. 4. User-Survey: Top-N Rated Movies Recommender System

B. Evaluation of Collaborative Filtering Movie Recommender Systems

Other inputs (year-range and movie title with release year) were also kept fixed during the evaluation of both collaborative filtering movie recommender systems, namely the Top-N Similar Movies Recommender System (by Cosine Similarity) and the Top-N Similar Movies Recommender System (by Pearson Correlation). This would provide consistency when comparing the results of both collaborative filtering movie recommender systems. The results of the users' survey for the Top-N Similar Movies Recommender System (by Cosine Similarity) and Top-N Similar Movies Recommender System (by Pearson Correlation) are shown as a bar-plot in Fig. 5 and Fig. 6, respectively.

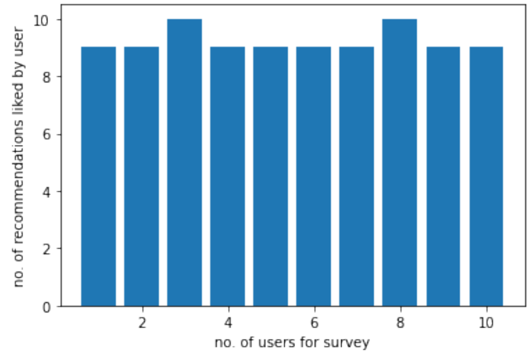

Fig. 5. User-Survey: Top-N Similar Movies Recommender System (by Cosine Similarity)

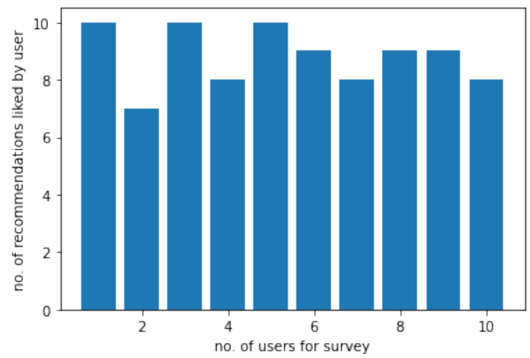

Fig. 6. User-Survey: Top-N Similar Movies Recommender System (by Pearson Correlation)

VI. DISCUSSION

A. Comparison between Content-Based Movie Recommender Systems

According to the results of the user survey, the Top-N Popular Movies Recommender System is far more effective than the Top-N Rated Movies Recommender System. Only three people liked less than nine out of ten movies recommended by the Top-N Popular Movies Recommender System, while three users liked nine or more out of ten movies recommended by the Top-N Rated Movies Recommender System. This indicates that individuals prefer to watch movies that are very popular (watched by a large number of people) rather than movies that are highly rated.

B. Comparison between Collaborative Filtering Movie Recommender Systems

The user survey results show that the Top-N Similar Movies Recommender System (by Cosine Similarity) is far more effective than the Top-N Similar Movies Recommender System (by Pearson Correlation). All users liked at least nine out of ten movies recommended by the Top-N Popular Similar Movies Recommender System (by Cosine Similarity), whereas six users liked nine or more out of ten movies recommended by the Top-N Similar Movies Recommender System (by Pearson Correlation). This demonstrates that both similarity measures are effective at recommending similar movies to the user, but the movies recommended by Cosine Similarity are more effective and better than the movies recommended by Pearson Correlation.

VII. CONCLUSION

This research study introduced the key concepts of recommendation systems. A concise overview of the several algorithms employed by Netflix, a popular video streaming platform, was accomplished. The hybrid model proposed here is implemented by combining content-based and collaborative filtering recommendation approaches. While content-based movie recommender systems use movie information to make recommendations, collaborative filtering uses the similarity measure of user ratings to provide recommendations. Furthermore, the implementation of both content-based movie recommender systems (i.e., Top-N Popular Movies and Top-N Rated Movies) and collaborative filtering movie recommender systems (i.e., Top-N Similar Movies using Cosine Similarity and Pearson Correlation) were extensively covered. All four models were evaluated via user surveys, in which each user rated how many movies he or she preferred from a given set of movie recommendations. The results show that movies recommended on the basis of popularity and similarity using the cosine similarity score were favoured by a significantly higher number of users than movies recommended by the other two models.

ACKNOWLEDGMENT

I would like to earnestly acknowledge the sincere efforts and valuable time given by Dr. Trevor M. Tomesh. His valuable guidance and feedback helped me to complete the research project successfully.

I would also like to thank all of the users (friends and family members) who participated in the survey. The information gathered from these user surveys was very helpful in the discussion and conclusion of the research project.

REFERENCES

[1] J. A. Konstan, and J. Riedl, Recommender systems: from algorithms to user experience, User Model. and User-adapt Interact., vol. 22, no. 1–2, pp. 101–123, 2012.

[2] F. Ricci, L. Rokach, and B. Shapira, Eds., Recommender systems handbook, 2nd ed. New York, NY: Springer, 2015, ch. 1.

[3] S. Wattal, Y. Hong, M. Mandviwalla, and A. Jain, Technology diffusion in the society: Analyzing digital divide in the context of social class, in 2011 44th Hawaii Int. Conf. on System Sciences, 2011.

[4] P. Melville, and V. Sindhwani, Recommender Systems, in Encyclopedia of Machine Learning and Data Mining, Boston, MA: Springer US, 2017, pp. 1056–1066.

[5] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, Collaborative Filtering Recommender Systems, in The Adaptive Web, Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 291–324.

[6] S. Kumar, K. De, and P. P. Roy, Movie recommendation system using sentiment analysis from microblogging data, IEEE Trans. Comput. Soc. Syst., vol. 7, no. 4, pp. 915–923, 2020.

[7] S. Agrawal, and P. Jain, An improved approach for movie recommendation system, in 2017 Int. Conf. on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), 2017.

[8] R. Hande, A. Gutti, K. Shah, J. Gandhi, and V. Kamtikar, Moviemender- A Movie Recommender System, Int. Journal of Eng. Sci. and Res. Tech., vol. 5, no. 11, 2016.

[9] C. A. Gomez-Uribe, and N. Hunt, The Netflix recommender system: Algorithms, business value, and innovation, ACM Trans. Manage. Inf. Syst., vol. 6, no. 4, pp. 1–19, 2016.

[10] P. G. Padti, K. Hegde, and P. Kumar, Hybrid Movie Recommender System, Int. Journal of Res. in Eng., Sci. and Manage., vol. 4, no. 7, pp. 311–314, 2021.

[11] A. Kashyap, S. B, S. Srivastava, A. PH, and A. J. Shah, A Movie Recommender System: MOVREC using Machine Learning Techniques, Int. Journal of Eng. Sci. and Comput., vol. 10, no. 6, pp. 26195–26200, 2020.

[12] B. Bhatt, P. J. Patel, and H. Gaudani, A Review Paper on Machine Learning Based Recommendation System, Int. Journal of Eng. Develop. and Res., vol. 2, no. 4, pp. 3955–3961, 2014.

[13] M. Goyani, and N. Chaurasiya, A Review of Movie Recommendation System, ELCVIA: Electron. Lett. on Comp. Vision and Image Anal., vol. 19, no. 3, pp. 18-37, 2020.

[14] F. M. Harper and J. A. Konstan, The MovieLens Datasets: History and Context, ACM Trans. on Interact. Intell. Syst., vol. 5, no. 4, pp. 1–19, 2016.

SUPPORTING MATERIALS

A. Examples of recommendations results by each movie recommender system

```
get_top_popular_recommendations(10, 2011, 2015, 'action')
Top-10 Popular Action Movies from 2011 to 2015 recommended for you:
```

	title	ratings_avg	ratings_count
1	Django Unchained (2012)	4.004882	20687
2	Dark Knight Rises, The (2012)	3.971349	19912
3	Avengers, The (2012)	3.780247	17572
4	Guardians of the Galaxy (2014)	3.932247	16324
5	The Hunger Games (2012)	3.518994	13741
6	Mad Max: Fury Road (2015)	3.855034	13479
7	Harry Potter and the Deathly Hallows: Part 2 (...	3.906986	13455
8	Star Wars: Episode VII - The Force Awakens (2015)	3.739115	12678
9	Edge of Tomorrow (2014)	3.940080	12425
10	Gravity (2013)	3.617050	12264

Fig. 1. Example of recommendations result provided by Top-N Popular Movies Recommender System

```
get_top_rated_recommendations(10, 2011, 2015, 'action')
Top-10 Rated Action Movies from 2011 to 2015 recommended for you:
```

	title	ratings_avg	ratings_count
1	Django Unchained (2012)	4.004882	20687
2	Dark Knight Rises, The (2012)	3.971349	19912
3	Edge of Tomorrow (2014)	3.940080	12425
4	Guardians of the Galaxy (2014)	3.932247	16324
5	Harry Potter and the Deathly Hallows: Part 2 (...	3.906986	13455
6	Big Hero 6 (2014)	3.879613	10379
7	Rush (2013)	3.867236	3574
8	Mad Max: Fury Road (2015)	3.855034	13479
9	Headhunters (Hodejegerne) (2011)	3.807249	1214
10	Kingsman: The Secret Service (2015)	3.798285	9620

Fig. 2. Example of recommendations result provided by Top-N Rated Movies Recommender System

```
get_top_similar_recommendations(10, 2005, 2015, 'Inception (2010)')
Users who watched Inception (2010) also watched these movies released in 2005-2015:
```

	title	similarity_score	ratings_count
1	Dark Knight, The (2008)	0.727263	149
2	Inglourious Basterds (2009)	0.646103	88
3	Shutter Island (2010)	0.617736	67
4	Dark Knight Rises, The (2012)	0.617504	76
5	Interstellar (2014)	0.608150	73
6	Up (2009)	0.606173	105
7	Avengers, The (2012)	0.586504	69
8	Django Unchained (2012)	0.581342	71
9	Departed, The (2006)	0.580849	107
10	Iron Man (2008)	0.572546	94

Fig. 3. Example of recommendations result provided by Top-N Similar Movies Recommender System (by Cosine Similarity)

```
get_top_similar_recommendations(10, 2005, 2015, 'Inception (2010)')
```

```
Users who watched Inception (2010) also watched these movies released in 2005-2015:
```

	title	correlation_score	ratings_count
1	Dark Knight Rises, The (2012)	0.601706	76
2	Prestige, The (2006)	0.587538	90
3	Batman Begins (2005)	0.523953	116
4	Hangover, The (2009)	0.507722	76
5	Interstellar (2014)	0.491152	73
6	V for Vendetta (2006)	0.454120	100
7	Slumdog Millionaire (2008)	0.445741	71
8	Ratatouille (2007)	0.421111	72
9	Departed, The (2006)	0.414245	107
10	Dark Knight, The (2008)	0.412281	149

Fig. 4. Example of recommendations result provided by Top-N Similar Movies Recommender System (by Pearson Correlation)

B. Tables of user surveys for each movie recommender system

user	from_year	to_year	genre	popular (liked/10)	rated (liked/10)
1	2011	2015	action	10	9
2	1991	2000	romance	7	7
3	2005	2015	mystery	10	8
4	2001	2015	fantasy	9	6
5	2011	2019	sci-fi	10	10
6	2005	2015	animation	10	5
7	1994	2019	war	8	8
8	2011	2019	adventure	10	9
9	2011	2019	horror	8	6
10	2005	2010	thriller	9	8

TABLE I

USER SURVEY FOR CONTENT-BASED MOVIE RECOMMENDER SYSTEMS

user	from_year	to_year	movie	similar-cosine (liked/10)	similar-pearson (liked/10)
1	2005	2015	Inception	9	10
2	2000	2018	Rataouille	9	7
3	1994	2004	Ocean's Eleven	10	10
4	2000	2018	Gladiator	9	8
5	2011	2018	The Conjuring	9	10
6	2012	2018	Deadpool	9	9
7	2008	2018	The Avengers	9	8
8	1994	2000	Fight Club	10	9
9	1991	2000	Pulp Fiction	9	9
10	2001	2018	La La Land	9	8

TABLE II

USER SURVEY FOR COLLABORATIVE FILTERING MOVIE RECOMMENDER SYSTEMS