Social Affinity-Based Group Recommender System

Minsung Hong[1], Jason J. Jung[1(✉)], and Minchang Lee[2]

[1] Department of Computer Engineering, Chung-Ang University, Heukseok, Seoul 156-756, South Korea
minsung.holdtime@gmail.com, j2jung@gmail.com

[2] Division of Public Administration and Welfare, Chosun University, Gwangju, Republic of Korea
savio@chosun.ac.kr

Abstract. Information collected from the social network is recently used to improve a performance of recommender systems to an individual user or a group. During selecting the items among the group members, the relationships (e.g., position, dependency, and the strength of the social ties) often has an important role than the individual preference in the group. Hence, we propose a novel recommendation method based on social affinity between two users. This recommendation method consists of (*i*) the similarity calculation between movies based on weighted feature, (*ii*) the generation of initial affinity network graph, and (*iii*) the computation of user's affinity to group based on the graph. Experimental results on synthetic dataset show that our proposed method can discover social affinities efficiently.

Keywords: Social affinity · Similarity · Weighted features · Group recommender system

1 Introduction

In this work, we focus on social interactions among users. People often resort to friends in their social networks for advice before purchasing a product or consuming a service [1]. The information collected from the social network is recently used to improve a performance of recommender systems to an individual user or a group.

Since these studies do not consider the real situations, we want to show a typical scenario. Suppose that there are two couples who try to select the best movies for watching. While the first couple has usually chosen the movies which a girl friend has preferred, the second couple has selected the movies which both have preferred as much as possible. Thus, during selecting the items among the group members, the relationships (e.g., position, dependency, and the strength of the social ties) often has an important role than the individual preference in the group.

P.C. Vinh and V. Alagar (Eds.): ICCASA 2015, LNICST 165, pp. 111–121, 2016.
DOI: 10.1007/978-3-319-29236-6_12

Hence, we propose the novel recommendation method based on social affinity between two users. Particularly, the similarity calculation among items, weight of item feature based on TF-IDF, and the graph based method are exploited to get the affinities among users who have not watched the movie together.

2 Related Work

Online social networks present new opportunities as to further improve the accuracy of recommender systems. In real life, people often resort to friends in their social networks for advice before purchasing a product or consuming a service [1]. Hence, some systems use the social information such as trust, and relation between users to improve the performance of recommender systems. Yang et al. [2] focused the "Friends Circles" which refines the domain-oblivious "Friends" concept. To influence the multiple domain specific, they proposed circle-based recommender system using the trust circles. Ma at el. introduced a Social recommendation (SoRec) model to adapted the social trust in recommender system in [3]. They used the directional concepts which are defined as an out degree (i.e., the number of users who a target user follows/trusts) and an in degree (i.e., the number of users who follow/trust target user). Jamali and Ester proposed a matrix factorization based approach for recommendation in social networks [4]. They incorporated the mechanism of trust propagation into a social model. For this mechanism, the model extracts a transitivity of trust in social network, as the dependence of a user on the direct neighbors. It can propagate to make a user's feature vector dependent on possibly all users in the network with decaying weights for more distant users.

Most work on recommender systems focus on the recommendation items to individual users. For instance, they may select a book for a particular user to read based on a model of that users preferences in the past [5]. However, we should consider the recommendation to group in many real recommendations such as a music of gym or health center, a TV program sequence for family, a travel destination with friends, and a good restaurant for colleagues to have a lunch, and so on. Some researches focused the aggregation of the individual preferences or other information for group. Masthoff [6] summarized eleven aggregation strategies of the individual user's likes and dislikes inspired by social choice theory, such as average, multiplicative, least miserty, fairness, and so on. Amer-Yahia et al. [7] proposed a recommender that aggregates preference of members based on member's relevance to create the recommendation for group. Then, they analyzed the preference disagreements between pairs of individuals and employed to rank the recommended list. Additionally, Kim et al. [8] taken into account both the effectiveness and the satisfaction of individual members to group recommendation. Their system generates a recommended book list and removes an irrelevant items in order to improve satisfaction of individual members. Finally, Quijano-Sanchez et al. [9] adopted the users' personality in the group and the trust of connections among members as the factors which improve a prediction accuracy of group's rating.

However, these recommender systems do not consider important factor which is called "affinity" such as position, leverage, and relationship in real world, like previous mention. In other words, it is the affinity of users about group. Furthermore, these need the preference of individual users to aggregate into group's preference. While, our method use item's features and history instead of it to calculate the affinity between two users. Also, this method can get the affinity based on graph, even though users had not watched the movie together.

3 Measuring Social Affinity Between Two Users

In this section, we describe the initial social affinity calculation to generate the social affinity network graph. It is divided by the calculation of the weighted feature to similarity, computation of the movie similarity, and the generation of affinity between two users based on the movie similarity.

3.1 Movie Features and Preprocessing

Before the similarity calculation, we account the used movie features in this paper. Debnath et al. [10] defined the distance of the 13 movie features which are served by IMDB[1]. Also, they analyzed the importance degree of features to choose the movie by users based on a linear regression equations. However, they supposed that a director feature has one people and equally applied the analysis results to all users. In case of both "Dumb and Dumber To" and "Crazy, Stupid, Love.", these movies had two directors. Besides, the same value inappropriate to all users. Therefore, we need to modify the features as following Table 1. Also, we have to use the novel weight calculation method which differently creates the weights according to users based on the basic idea of TF-IDF. In this paper, we use the 7 features such as release, rating, director, genre, leading actor, country, and leading actor, country, language (note that used rating is collected by the IMDB and the leading actors extracted from the Naver movie[2]).

To analysis the similarity between movies, many methods are studied. We adopt the Jaccard's, Overlap coefficient, and Euclidean distance as the similarity measure. We need to transform the feature values into terms, because former two methods use a common ground between two set, we use the TF-IDF to create the weight of each feature, and we avoid a duplication among features (e.g., D_Seth MacFarlane, LA_Seth MacFarlane). Also, we apply prior Table 1 to the Euclidean distance.

3.2 Weighted Feature Based on TF-IDF

Above-mentioned weighted features which is studied by Debnath is not perfect to users group in aspects of personalization. Hence, we propose the method which

[1] IMDB, http://www.imdb.com
[2] NAVER MOVIE, http://movie.naver.com/

Table 1. Features used in movie recommendation

Feature	Type	Cardinality	Distance measure
Release	Year	YYYY	$(300 - \lvert Y_1 - Y_2\rvert)/300$
Rating	Float	(0 – 10)	$(10 - \lvert R_1 - R_2\rvert)/10$
Director	(String)*	$(< name >)^*$	$(\lvert D_1 \cap D_2\rvert)/D_{max}$
Genre	(String)*	$(< genre >)^*$	$(\lvert G_1 \cap G_2\rvert)/G_{max}$
Leading actor	(String)*	$(< name >)^*$	$(\lvert LA_1 \cap LA_2\rvert)/LA_{max}$
Country	(String)*	$(< country >)^*$	$(\lvert C_1 \cap C_2\rvert)/C_{max}$
Language	(String)*	$(< language >)^*$	$(\lvert L_1 \cap L_2\rvert)/L_{max}$

calculate the weight of feature based on TF-IDF. The TFIDF, short for term frequency inverse document frequency, is a numerical statistic that is intended to reflect how important a word is to a document in a collection or corpus. The term frequency express the importance degree in a target document. While, the inverse document frequency is a measure of how much information the word provides, that is, whether the term is common or rare across all documents [11]. To use it in creation of feature's weight, we need some basic defines and setting.

Target document(d) is a set of the feature values about movies which are seen together by both users.
Document set(D) represent the sets of the feature values of movies which are watched by each users.
Term(t) appear terms (the feature value of movies) in the *Target document*(d).

Next, we describe the proposed weight calculation method using prior example which is the movie history between "Fames" and "John". In this case, the d and t is an enumeration of the feature values both "Interstellar" with "The Wolf of Wall Street" movie and the it's terms, respectively. Also, the D include the movies ("Inception", "The Dark Knight", "Dumb and Dumber To", "Ted", "The Mask", and "Crazy, Stupid, Love.") as documents. We use the "raw frequency" as TF weighting scheme and TF-IDF integration method as Definition 1. The different with basic TF-IDF is square of the TF part. The document which is set by use in this domain is different with normal documents, because it is a list of feature value. Therefore, the count of term in the document d is very small. However, if the movies which is watched together between two users have the same feature values, it have to deal with as very importance point to them. To reflect this actual state, we adopt the square of TF in this study. Also, λ_i is normalization constant of each feature. This weight is combined with coefficient (or distance) of each feature to calculate the similarity between two movies.

Definition 1 (TF-IDF for Feature Weight). *Let $d \in D$ and $f \in Feature$, given term $t \in d$. The weight between two users based on TF-IDF is defined as follows:*

$$tf(t,d) = f_{t,d}, \quad idf(t,D) = log\frac{N}{1 + |\{d \in D : t \in d\}|}, \tag{1}$$

$$tf\_idf(t,d,D) = tf(t,d)^2 \times idf(t,D), \tag{2}$$

$$weight(James, John) = \lambda_f \times tf\_idf(t,d,D). \tag{3}$$

3.3 Movie Similarity Based on the Weighted Feature

Previously, we refer to the used the similarity calculation methods between two movies such as Jaccard's coefficient, Overlap coefficient, Euclidean distance. In this chapter, we respectively show these process through one part in prior movie history.

Jaccard's and Overlap create the similarity based on a common attribute like Definition 2. Also, Euclidean method uses the definitions in Table 1. These methods are applied to $each feature$ as Definition 3. In the next section, this similarity is altered to the affinity between two users.

Definition 2 (Jaccard's and Overlap coefficient). *The similarity using the Jaccard's and Overlap coefficient between two movies is defined as follows:*

$$jaccrard(a,b) = \frac{N_{a \cap b}}{N_{a \cup b}}, \tag{4}$$

$$overlap(a,b) = \frac{N_{a \cap b}}{Min(N_a, N_b)}, \tag{5}$$

$$Euclidean(a,b) = Distance(a,b) \quad in\ the\ Table\ 1. \tag{6}$$

Definition 3 (Similarity between two movies). *Let $d \in D$, the w_f, s_f are the weight based on TF-IDF and the similarity which is obtained by Jaccard's coefficient (or Overlap, Euclidean), respectively. Also, the f is 7 as the number of the features. The similarity between two movies is integrated with weight by definition as follows:*

$$Similarity(w_f, s_f) = \sum w_f \times s_f, \quad f \in Features. \tag{7}$$

3.4 Social Affinity Based on Movie Similarity

Until now, we calculate the similarity through six methods used both the similarity measures such as Jaccard's, Overlap coefficient, and Euclidean Distance with the weighted features. The social information in our recommender system is the affinity between two users in case of the view movie together them. To adopt it in recommendation, we need to transform the similarities as the affinities between two users int Definition 4.

Definition 4 (Affinity between two users). *Let user A and B is the target users. Also, the k of MA and the l of MB appear the movie lists which are watched by user A and B, respectively. The m of MC expresses the list of movies which are viewed by them together. In this case, the affinity between A and B is created by the similarity which is obtained in presence as follows (we mark the Definition 3 into $sim()$ and $affinity()$ into $affi()$ by the limitation of length):*

$$affi(A,B) = \begin{cases} \frac{\sum_{i,j=1}^{i=k,j=m} sim(MA_i,MC_j)/k}{\sum_{i,j=1}^{i=k,j=m} sim(MA_i,MC_j)/k + \sum_{i,j=1}^{i=l,j=m} sim(MB_i,MC_j)/l} & if\ m \neq 0, \\ 0.5 & if\ m = 0. \end{cases} \quad (8)$$

4 Exploiting Social Affinity Graph to Group Recommendation

In this section, we account the recommendation of group through the request. Firstly, we introduce the calculation of social affinity between two users based on the social affinity graph. Then, the computation of the user's social affinity to group. Finally, we show the process of the proposed SAGRS.

4.1 Social Affinity Between Two Users on the Network Graph

Several existing work try to improve the performance of recommender system based on social information. Jiaming and Wesley [12] proved the effect of "Immediate Friend" (i.e., the friends who are directly connected in social network) and "Distant Friend" (i.e., the friends who have the indirect connection such as two or three hop in social network) in recommendation. Ma et al. applied the concepts "Follower" and "Leader" in recommender system [3]. In this paper, we reflect these concepts to generate the affinity between two users who watch the movie together or not.

1. Proposed affinity has the direction between two users as the concept "Follower" and "Leader",
2. The multiple hop include the "Immediate friend" and "Distance friend" in the social affinity network.

To describe a using reason of this concept, a simple scenario is shown as follows:

> "Robert" and "John" had watched the movies with "Patricia" and "James", respectively. In this case, "James" can look for advice by "Robert" and "John" to see the movie with "Patricia".

That is, user can refer the hints of his/her friends to the preference of target user, even though he/she doesn't have experience which look the movie together.

To understand this situation, we add some suppositions into this scenario. When "James" watch with "John", he has a effect to "John", and "John" has the similar affinity with "Patricia". Besides, "James" has alike influence with "Robert", and "Robert" has a big leverage to "Patricia". In this case, we can expect that the affinity of "James" is bigger than "Patricia". Likewise, we can obtain the information about the affinity from indirect connection, even though between user are not directly connected. To consider this point into recommendation, we express the affinity among users into graph. The vertex of graph appears the user, and it's edges have direction as the affinity of user pair. Also, "James" is indirectly connected with "Patricia" and this connection is called "two hop". Then, the affinity between two users is calculated by Definition 5. It is comprised of the creation of affinity and normalization.

Definition 5 (Affinity Formula for Recommendation). *Let H is the maximum number of indirect connection in the affinity network graph, and P is the number of path in each hop such as one hop, two hop, three hop. The affinity for recommendation is defined as follows:*

$$affinity(A, B) = affi(A, B) + \sum_{h=2}^{H}\sum_{p=1}^{P}(product(aff_1, ..., aff_h)/P), \tag{9}$$

$$Affinity(A, B) = \frac{affinity(A, B)}{affinity(A, B) + affinity(B, A)}. \tag{10}$$

We can get the "Affinity" between "James" and "Patricia" using the Affinities of indirect connections (i.e., two hop about "James, John, and Patricia" and "James, Robert, and Patricia", three hop about "James, Mary, Michael, and Patricia". This affinity is calculated by the follows formula.

In the affinity, First term express the initial affinity about direct connection, and Second and third term appear the indirect connection as two and three hop, respectively. Particularly, Second term which is a multiply between two affinities regular bigger than third which is a product of three values, because the rage of used figures less than or equal to 1. It can be larger an influence of near connection than a far connection.

4.2 Social Affinity of the Users to Group

Until now, we explain the generation of initial affinity graph based on the affinity between two users who had watched the movie together. Additionally, we introduce the calculation of the affinity between users who had not watched the movie together based on the indirect connection in the affinity graph. We note that It use the values in the initial affinity graph. To answer our third research question, from now on we describe the member's affinity computation method to group which has three or more users. For this method, we use the case of the group which has members such as *James*, *John*, *Patricia*, and *Robert*. It appears the affinities which are calculated by the indirect connection among group members. Also, the affinities nearby an user express the affinities of the user about the others.

In this case, we can consider several methods which get the affinity of users to group such as average, product, and so on. However, the information of high affinity can dilute in the integration methods. Therefore, we adopt the maximum method which used the largest value as the user's affinity about the group which is processed as follows:

1. a maximum affinity and an applicable user is found,
2. the affinities relate to the user is removed,
3. process 1, 2 are repeated before a final user,
4. the affinity of the final user is allocated as the affinity about an user of the previous step,
5. all affinities are normalized to group.

"Robert" who has large affinities about the others has higher affinity to group than other users as 0.333. While, In the case of the "John" and "Patricia", they relatively have low affinity to group. It show the propriety of our method in the example. Until now, we get the affinity of users to group based on the affinity between users. In next section, we explain the proposed SAGRS based on the previous definitions.

4.3 Recommendation Based on the Social Affinity

The SAGRS is comprised of three parts:

1. In the case of the input users movie history, this information is transformed and saved to database in the preprocessing step. The transformation makes the terms about the feature values of the movies to calculate the weighted based similarity. The values is served by IMDB and Naver movie. We skip the detail description of this part, because it is explained in Sect. 3.
2. Initial affinity graph is generated based on the user's movie history. It not influences in the aspects of the recommendation time, because it can be processed in idle time. Alike the preprocessing part, it is introduced in Sect. 3. Therefore, we show the simple order of this part.
3. In the recommendation part, the movie list is generated to recommendation through the request of users. Firstly, the affinities of the users who are belong to the request group are computed by the initial affinity graph. Secondly, the affinities of users to group are created by these. Finally, these are applied the recommendation for the group. It's detail context is described in the next section.

Social Affinity Based Group Movie Recommendation. The first, second part are process in the idle time. While, third part is started by the recommendation request of the users. This structure makes to reduce the response time of recommendation. From now on, we explain the steps of third part as:

1. The users who request the movie list are found in the graph.
2. The affinities of the pair users in the group are calculated by indirect connection in the graph.
3. The affinities of each user are computed about the group.
4. The movie lists are created by similarity to each user.
5. The user's affinities are integrated with the similarities of the movie lists as values to one movie list.
6. The list is ranked by the unified values, and a top n movies are recommended to the group.

Because the others are described in the previous sections, we want to explain steps 4, 5, and 6. Firstly, the movie lists for each user are obtained by the similarity of the overlap coefficient between the user's movie history and the novel movies, because the similarity of overlap properly expresses the affinity between two users than the others. Then, the product of movie similarities and the affinity of user are unified as one movie list to the top N movie recommendation.

5 Experiment

To evaluate the proposed SAGRS, we create the virtual users and movie history in this section. Firstly, the users are divided by three groups (i.e., group A is very certain, group B is certain, and group C is uncertain) based on the degree of obviousness. Hence, we generate the 18 users and distribute users into three groups. Also, the characteristics of users who belong to group A and B are set. The 100 movies of which the feature information is collected from the IMDB and the Naver movie as the 7 features such as release, rating, director, genre, leading actor, country, and language. Then, these are allocated as 7 to 10 movies into per user, because the movie which are watched together is overlap. Also, we use the graph which are made by the virtual movie history. The users of the group A are appeared by black circle and white name as a vertex in the graph. The users of the group B, C are expressed by the gray circle and the white circle as black name, respectively. The average and standard deviation of the virtual data result among group A, B, C are shown in Table 2. In this result, the proposed method which use the affinity between users is effective, because it is natural as our setting (i.e., the affinity of the group A bigger than group B, the group B larger than the group C).

Table 2. The affinities of the result among groups

Groups	Average of affinity	Groups	Average of affinity	Standard deviation
A to B	0.531	B to A	0.469	0.022
B to C	0.580	C to B	0.420	0.035
A to C	0.584	C to A	0.416	0.055

6 Conclusions and Future Work

The existing works adopted the friends information of user in online social networks to recommender system. In this paper, we propose the method which use the affinity between users based on the movie watching history among users. Besides, it can operate without the user's rating or other profile information. Also, we show the validation of the method using the virtual users and movie watching history.
However, the limitations of this study are as follows:

1. We need to test on the more large user set instead of the only 18 users.
2. We have to create the virtual network which reflect the real world based on the normal theory such as scale-free network, small wold network, and random network.
3. The various networks based on the theories are tested about the range of the used indirect connection (i.e., mutiple hop) in the aspects of the data sparsity.
4. Above all, our method needs the revaluation using the survey of user, because the experiment is progressed on the virtual data which are the users and movie history of three group based on the degree of obviousness.

Hence, we plan to construct the real system which connect to *Facebook*[3] based on the proposed SAGRS. Through the it's service, we can analyze the performance of the our system such as satisfaction, usefulness, response time, and so on.

Acknowledgement. This research was supported by the MSIP (Ministry of Science, ICT and Future Planning), Korea, under the ITRC (Information Technology Research Center) support program (IITP-2015-H8501-15-1018) supervised by the IITP (Institute for Information &communications Technology Promotion). Also, this work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIP) (NRF-2014R1A2A2A05007154). Also, this research was supported by SW Master's course of hiring contract Program grant funded by the Ministry of Science, ICT and Future Planning (H0116-15-1013).

References

1. Yang, X., Guo, Y., Liu, Y., Steck, H.: A survey of collaborative filtering based social recommender systems. Comput. Commun. **41**, 1–10 (2014)
2. Yang, X., Steck, H., Liu, Y.: Circle-based recommendation in online social networks. In: Yang, Q., Agarwal, D., Pei, J., (eds.) KDD, pp. 1267–1275. ACM (2012)
3. Ma, H., Yang, H., Lyu, M.R., King, I.: Sorec: social recommendation using probabilistic matrix factorization. In: Shanahan, J.G., Amer-Yahia, S., Manolescu, I., Zhang, Y., Evans, D.A., Kolcz, A., Choi, K.S., Chowdhury, A., (eds.) CIKM, pp. 931–940. ACM (2008)
4. Jamali, M., Ester, M.: A matrix factorization technique with trust propagation for recommendation in social networks. In: Amatriain, X., Torrens, M., Resnick, P., Zanker, M., (eds.) RecSys, pp. 135–142. ACM (2010)

[3] Facebook, https://www.facebook.com/

5. Masthoff, J.: Group recommender systems: combining individual models. In: Ricci, F., Rokach, L., Shapira, B., Kantor, P.B. (eds.) Recommender Systems Handbook, pp. 677–702. Springer, US (2011)
6. Masthoff, J.: Group modeling: selecting a sequence of television items to suit a group of viewers. User Model. User-Adapt. Inter. **14**(1), 37–85 (2004)
7. Amer-Yahia, S., Roy, S.B., Chawla, A., Das, G., Yu, C.: Group recommendation: semantics and efficiency. PVLDB **2**(1), 754–765 (2009)
8. Kim, J.K., Kim, H.K., Oh, H.Y., Ryu, Y.U.: A group recommendation system for online communities. Int. J. Inf. Manage. **30**(3), 212–219 (2010)
9. Sánchez, L.Q., Recio-García, J.A., Díaz-Agudo, B., Jiménez-Díaz, G.: Social factors in group recommender systems. ACM TIST **4**(1), 8 (2013)
10. Debnath, S., Ganguly, N., Mitra, P.: Feature weighting in content based recommendation system using social network analysis. In: Huai, J., Chen, R., Hon, H.W., Liu, Y., Ma, W.Y., Tomkins, A., Zhang, X., (eds.) WWW, pp. 1041–1042. ACM (2008)
11. Wikipedia: Tf-idf. https://en.wikipedia.org/wiki/Tf
12. He, J., Chu, W.W.: A social network-based recommender system (SNRS). In: Memon, N., Xu, J.J., Hicks, D.L., Chen, H. (eds.) Data Mining for Social Network Data. Annals of Information Systems, vol. 12, pp. 47–74. Springer, US (2010)