MovMe: Personalized Movie Information Retrieval

Hyung W. Kim, Kee J. Han, Minsam Ko, Mun Y. Yi
KAIST/Department of Knowledge Service Engineering, Daejeon, Republic of Korea
Email: {hw_kim, ham1212, msko, munyi}@kaist.ac.kr

***Abstract*— MovMe is a personalized movie information retrieval method that considers user comments and their ratings about movies. The central idea of MovMe is that a user's movie preference can be identified by analyzing their past rating patterns and comments about movies. The performance of MovMe was compared with that of the other two methods based on tf-idf and user ratings. Using the online movie data available from Naver Movie, which is the largest movie portal in Korea, we ran an experiment in which 600 queries (20 queries for each of 30 users) were run to retrieve movies and the three methods were compared. The test results based on the mean values of user ratings of the first five movies from a list of movies retrieved by each retrieval method shows that MovMe is a better retrieval method than the other two methods in providing satisfying search results.**

***Index Terms*—Personalized information retrieval, User comments, Web services, User satisfaction, Movie recommendation**

I. Introduction

As the Web is growing each day with new pages and contents, the provision of satisfying search results is a crucial issue. The conventional method to measure the quality of search result is to measure recall and precision of documents returned by a user query. In case of content search such as finding movies, books, or sales items, however, recall and precision are not fully capable of evaluating search quality because, in that case, user satisfaction also is an important factor, which is not captured by those criteria.

Due to this importance of the degree of personal satisfaction in content search, researchers have been conducting various studies of reflecting someone's preferences towards contents into search results. Recommender systems are one of the examples to analyze preferences of all the users by using user profiles to recommend the most suitable items to each user. Furthermore, in personalized information retrieval area, information retrieved by a query is used for personalization by considering user's interests and preferences. However, these conventional personalization methods have several limitations. First, these methods only utilize users' previous historic data – in particular, user ratings. A user rating is a numerical summary of user evaluation and does not reveal the underlying information used for the evaluation. Second, there is a privacy concern that personal privacy might be violated while the Web applications collect user information for user-profiling.

The era of Web 2.0 has caused much more active user participations. Users create contents by interacting each other on the Web. For instance, people can share their opinions and emotions by posting a wall on Facebook and also can easily rate movies or books with a short paragraph of their opinion on E-bay or Youtube. These new user-created sources are created enormously each day thorough various channels on the Web. However, most of current information retrieval methods are not fully optimized to analyze these new types of information for improving personalized information retrieval. MovMe, proposed in this paper, is a personalized movie information retrieval method which utilizes not only numerical rating values but also textual data in which users commented about contents in order to search the most suitable movies that users had not yet seen.

IMDB and NAVER are two leading commercial movie search websites in the U.S.A and South Korea respectively. Their overall search process is that a user queries a term to the database and then the systems only return a list of the movies in which the title contains the given string. This method is commonly called "text matching search." On the basis of this text-matching search process, MovMe has three additional advancements as follows. First, MovMe is able to find movies although its title does not contain any terms in a query. For instance, the words such as 3D, twisted-ended, story are not usually included in movie titles, however, MovMe is capable of finding movies which are relevant to the words. Second, MovMe still carries advantages of personalized search because MovMe uses user's rating scores which reflect actual user's preferences on movies.
Third, MovMe does not bring up privacy issue because MovMe processes user-profiling by only extracting rating scores and text from user comments. These types of information are agreed to be read by others because the reason why the users post comments is not to hide but to share their opinions, thus, in our method privacy issue is not a limiting factor. Therefore, this paper will examine how MovMe performs relative to other commercial movie search websites which adopt conventional searching methods.

II. Related Work

Generally, "Personalization" means that providing the contents which correspond with the users' needs [1]. Personalized Information Retrieval can be divided into various types based upon the ways of generating user profiles and types of information to be personalized.

Personalized information retrieval generates user profile by 1) utilizing users' dynamic inputs, 2) using previous queries [2] and click-through analysis [3][4], and 3) analyzing users' social profile [5]. There also exist two ways of choosing right types of information to be personalized: 1) query expansion by re-weighting the original query or adding new terms to the query based on the users' interest [6] and 2) re-ranking and filtering of the search results using user profiles [7].

DOI: 03.CSS.2011.1.65

However, these conventional personalization methods have faced several challenges and difficulties in applying to contents search. First, these methods only consider users' previous history data rather than his/her actual preferences. Second, there is a privacy concern that users' personal privacy might be violated while the Web applications collect user information for user-profiling.

The purpose of recommender systems is to recommend items that users had not yet considered. Recommender systems can be generally divided into two types: collaborative filtering and content-based filtering [8]. Collaborative filtering recommends contents based upon user's similar tastes [9][10][11]. However, this system only considers users' numerical ratings so it is not easy to find out evaluation criteria of the users when they rate those contents. Unlike collaborative filtering, content-based filtering analyze user's past rating patterns to generate recommendation. In this system, target content is compared to other contents which the user rated in the past. If the content is similar to other contents which the user rated high in the past, the content is also highly expected to be preferred by the user [12][13].

Although content-based filtering is simple and convenient for recommendation process, collaborative filtering generally shows better performance than content-based filtering because of its much various recommendation sources such as preferences and similarities between users. Due to the fact that collaborative filtering only utilizes numerical values rated by users, however, it is limited to apply to personalized information retrieval which requires various aspects of each content. Other than recommender system, there are several studies conducted using user comments as a main source of experiment. Some studies show that the number of user comments posted on news [14] and blog posts [15] can be an indicator of popularity. From all of these user comments related-studies, in conclusion, user comments enable improvement of various areas in different ways [16][17].

III. MovMe

MovMe, proposed in this paper, is a personalized movie search system which utilizes numerical points rated by users and short textual data extracted from user comments. Each user's index is weighted based upon similarities between users who posted comments and their rating values. To do so, the system can generate optimized indexes to each user and search the movies which the users might like most. For the verification purpose, user comments are collected from Naver Movie and used to test MovMe's performance.

Before the detailed explanation about MovMe, it is crucial to understand the nature of user comments. Followings are detailed observations about user comments entered into the movie review site from which the experiment data were obtained.

A. Comments on Naver Movie

We first collected a large number of data set to be tested from Naver Movie (http://movie.naver.com), which is one of the most famous commercial websites providing a large volume of movie information and user comments in Korea. The collected data consist of movie descriptions (title, genre, credit, and synopsis) and user comments about movies released in the past five years. In Naver Movie, users can rate a movie in two ways, a numerical rating from 1 to 10 and a short review comment in 40 Korean characters (80 English characters). In the dataset, there were total 2269 movies and 2,189,989 comments entered by 883,583 users.

B. Quantity of User Comments

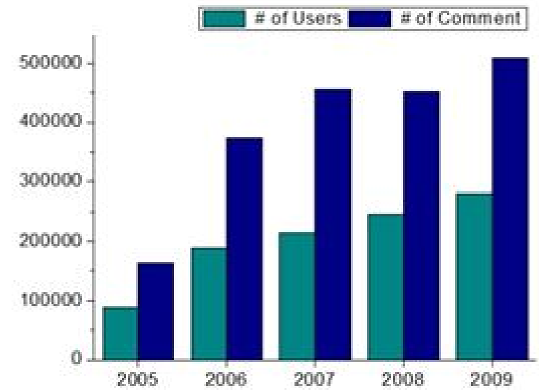

Figure 1. The number of users and comments from 2005 to 2009

As shown above, the number of comments increased every year. In Fig. 1, the number of user comments in 2009 tripled since 2005. In addition, the number of users posting user comments also increased considerably. In 2009, 281,078 users posted comments in total, while there were only 89,433 users in 2005.

C. Quality of User Comments

User comments can contain some useful information for improving the quality of personalized movie search. A page in Naver Movie consists of three parts; movie description, photos or ads related to the movie and user comments. Movie description is written by a service provider and it includes movie title and information about director, actors, genre, and story of the movie. Because this section has most of necessary data about movie, we assume that user comments can have meaningful data if user comments contain similar words to Movie description. To do so, user comments were investigated to see how many common words appeared both in the movie description and the user comments. From Table I, 60.05% of the words appeared in the movie description also appear in the user comments. This shows that the user comments likely include a certain number of meaningful words which can be used for personalized search. Then how can be these user comments utilized in personalized search? Indexes of each movie are created by combining conventional tf-idf with ratings scores and user similarities. If a user inputs a query on the system, then a personalized user index table

TABLE I.
The Number of Terms appeared in Descriptions and User Comments

	Descriptions	User Comments	Terms Appeared Both in Descriptions and User Comments
# of terms	15023	17158	10128 (60.05%)

DOI: 03.CSS.2011.1.65

returns a list of movies to the users. The equation below is the way of creating a movie index table for each user.

$$idf_a = \log \frac{|M|}{|\{m_j : t_i \in d_j\}|} \quad (1)$$

$$UserSim_{u,n} = \frac{\sum_{i \in R_{u,n}} r^*_{u,i} r^*_{n,i}}{\sqrt{\sum_{i \in R_u} {r^*_{u,i}}^2}\sqrt{\sum_{i \in R_n} {r^*_{n,i}}^2}} \quad (2)$$

Equation (1) is used for calculating idf values of words. In the first equation, the cardinality of M which is the total number of movies in the test set. $|\{m_j : t_i \in d_j\}|$ means the number of movies where the $term_a$ appears. Equation (2) is especially widely used in collaborative filtering to calculate similarities between users [8]. The range of this value lies between 0 and 1. As the value is getting closer to 1, two users have more similar preferences.

$$MovMe(User_u, Term_{(a,n) \in movie_i}) = Rating_{(a,n) \in movie_i} \times UserSim_{u,n} \times idf_a \quad (3)$$

$$MovMe(User_u, Term_{a \in movie_i}) = \sum_{n \in Movie_i} MovMe(User_u, Term_{a,n \in movie_i}) \quad (4)$$

Equation (3) is to calculate a weight of $Term_a$ among user comments posted by $User_n$ for $Movie_i$. Using (4), values of which appeared in all the user comments for are added. The value of of for is obtained. The rest values of all the term of are obtained as above and then finally index table of can be obtained as follow.

$$MovMe\ RS = \frac{Query \cdot MovMe(User_u, Movie_i)}{||Query| \times |MovMe(User_u, Movie_i)||} \quad (5)$$

When $User_u$ types a certain query, ranking score (RS) between each movie and the query can be computed by using (5). MovMe RS is cosine similarity between query and each movie. The movies are shown to the user from highest RS value to lowest RS value.

In the next section, MovMe which has been obtained by the above formulas will be compared with conventional tf-idf method and tf-idf & rating method, which only excludes user similarity from (3), to verify its performance.

IV. Experiments & Result

In this section, we discuss the performance of MovMe. For conducting a test, we selected a subset of the collected data because the original data set was too large. We reduced the data set by focusing on data created in the most recent year (i.e., year 2009). The test data consisted of 132,962 ratings and comments about 1,663 movies from 2,756 users for the year 2009. Among those 2,756 users, we specifically chose 30 users which have posted more than 100 ratings and comments about movies (i.e., heavy users) and also have high standard deviation of ratings (i.e., a wide range of variations).

To generate query terms that will be used for movie search, we recruited three people (i.e., college students in Korea) and asked them to provide 15 query terms that they use most often for movie search. After removing redundant terms, we selected 20 final terms randomly. Table II shows the 20 chosen query terms, which were used for the experiment. When these terms were queried, it was translated to Korean because our movie data set were written in Korea. In the experiment data set, each movie was already rated by the chosen 30 users.

TABLE I.
Total Twenty queries for performance evaluation

No.	Query	No.	Query
Query #1	Action	Query #11	Ending
Query #2	Touching	Query #12	Story
Query #3	Twist-ended	Query #13	Korea
Query #4	Director	Query #14	Actor
Query #5	Comedy	Query #15	3D
Query #6	War	Query #16	CG
Query #7	Recommend	Query #17	Happy
Query #8	Sorrow	Query #18	Fun
Query #9	Tear	Query #19	Peaceful
Query #10	Laugh	Query #20	Music

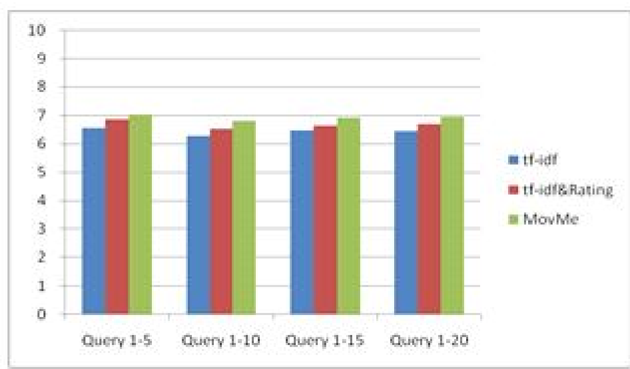

Figure 2. Summary of Mean Values based on the number of Queries

The performance of each method for indexing comments (1:tf-idf, 2:tf-idf & rating, 3:MovMe) was measured by calculating the mean values of past ratings of first 5 movies from the search list of movies returned by the search method. We decided to use the first 5 movies because users are likely to choose movies in the first search result page. In other words, if someone inputs a query on MovMe then MovMe returns a list of movies to him/her. Then only first 5 movies appeared from the list are chosen for calculating average movie score of the 5 movies. If the score is high enough, then we can verify that MovMe is successfully able to return similar types of movies which he/she liked in the past. Furthermore, if the mean value obtained by MovMe is higher than other mean values obtained by tf-idf and tf-idf & rating methods, then it could be strong evidence that MovMe performs better than other two methods.

DOI: 03.CSS.2011.1.65
ACEEE

Fig. 2 and Table III show the mean values of each method according to 5, 10, 15 and 20 numbers of queries. When the number of queries increases, the mean value of MovMe is constantly higher than other two methods. When 20 query terms are used, the final mean values of tf-idf, rf-idf & rating and MovMe are 6.450, 6.692, and 6.948 respectively. This result strongly implies that MovMe has constantly better performance than other methods regardless of the number of queries. Table IV shows how much MovMe performs better than other two methods when all the 20 queries are used. According to the paired t-test, mean difference between tf-idf & rating and tf-idf method is 0.241 and the difference between tf-idf&rating and MovMe is 0.257. These mean that tf-idf&rating method is able to return 0.241 higher scored movie than tf-idf method. Meahwhile,

TABLE. III
Mean of User Ratings by the Number of Queries

		Mean	Std. Deviation	Std. Error Mean
Query 1-5	tf-idf	6.556	2.106	0.077
	tf-idf & rating	6.864	2.093	0.076
	MovMe	7.044	2.102	0.077
Query 1-10	tf-idf	6.290	2.250	0.047
	tf-idf & rating	6.532	2.226	0.047
	MovMe	6.825	2.144	0.045
Query 1-15	tf-idf	6.468	2.280	0.042
	tf-idf & rating	6.637	2.232	0.041
	MovMe	6.912	2.150	0.039
Query 1-20	tf-idf	6.450	2.263	0.037
	tf-idf & rating	6.692	2.194	0.036
	MovMe	6.948	2.118	0.035

MovMe is able to search 0.257 higher-scored movies than tf-idf-rating method.

Finally the mean difference between MovMe and tf-idf method is 0.498. This also means that MovMe is able to search 0.498 higher scored movies than tf-idf method. Each of these results is statistically significant because all of its p-values are less than 0.001. As mean difference between MovMe and other methods is greater, MovMe performs better.

Conclusions

In this paper we propose MovMe, a search system that analyzes user comments and ratings from movie contents and utilizes them for personalized movie information retrieval. MovMe was tested with 30 chosen users of a large online movie review site in Korea and 20 query terms were collected from 3 people. The result of the experiment shows that the mean value of the past rating from the first 5 movies is 0.498 ($p<0.001$) points higher than that of the tf-idf method. This result implies that MovMe produces a list of higher satisfying movies than tf-idf method. MovMe has the following advantages. First, by using MovMe, current movie search engines based on exact title matching method can be improved as MovMe can retrieve related titles even though the title does not match the query term. Second, MovMe improves user satisfaction by considering preferences of each user.

Third, there is privacy violation is not a limiting factor for MovMe because the method uses comments and rating values, which are already meant to be shared.

TABLE. IV
Paired Differences Among Three Methods

	Paired Differences					t	df	Sig. (2-tailed)
				95% Confidence Interval of the Difference				
	Mean Diff	Std. Deviation	Std. Error Mean	Lower	Upper			
tf-idf - tf-idf & rating	-.241***	1.515	.0247	-.290	-.190	-9.757	3749	.000
tf-idf & rating - MovMe	-.257***	1.614	.0264	-.308	-.205	-9.729	3749	.000
tf-idf - MovMe	-.498***	1.396	.0228	-.543	-.453	-21.846	3749	.000

Acknoweldge

This work was supported by the National Research Foundation of Korea(NRF) grant funded by the Korea government(MEST) (No. 2011-0029185).

References

[1] C. Shahabi and Y.C. Chen. “Web Information Personalization : challenges and approaches,” In Proceedings of 3rd Workshop on Databases in Networked Information Systems (DNIS), 2003
[2] B. Tan, X. Shen, and C. Zhai, “Mining long-term search history to improve search accuracy,” In Proceedings of KDD, pp 718–723, ACM Press, 2006.
[3] T. Joachims, L. Granka, B. Pang, H. Hembrooke, and G. Gay, “Accurately interpreting clickthrough data as implicit feedback,” In Proceedings of SIGIR, pp 154–161, 2005.
[4] Z. Dou, R. Song, and J.-R. Wen. “A large-scale evaluation and analysis of personalized search strategies,” In Proceedings of WWW, pp 581–590, ACM, 2007.
[5] D. Carmel, N. Zwerdling, I. Guy, S. Ofek-Koifman, N. Har’el, I. Ronen, E. Uziel, S. Yogev and S. Chernov, “Personalized social search based on the user’s social network,” Proceedings of the 18th ACM Conference on Information and Knowledge Management (CIKM’09), ACM , 2009.

DOI: 03.CSS.2011.1.65

[6] P. A. Chirita, C. S. Firan, and W. Nejdl, "Personalized query expansion for the web," In Proceedings of SIGIR, pages 7–14ACM, 2007.

[7] X. Shen, B. Tan, and C. Zhai, "Implicit user modeling for personalized search," In Proceedings of CIKM, pp 824–831, ACM, 2005.

[8] G. Adomavicius, and A. Tuzhilin, "Toward the Next Generation of Recommender Systems: A survey of the State-of-the Art and Possible Extensions," IEEE Transactions on Knowledge and Data Engineering vol. 17, No. 3, 2005, pp 734-749.

[9] J.A. Konstan,, et al., "GroupLens: Applying Collaborative Filtering to Usenet News," Communications of the ACM vol. 40, no. 3, 1997, pp 77-87.

[10] G. Linden, B. Smith, and J. York, "Amazon.com Recommendations: Item-to-Item Collaborative Filtering," IEEE Internet Computing, Jan./Feb. 2003.

[11] B.N. Miller, et al., "MovieLens Unplugged: Experiences with an Occasionally Connected Recommender Systems," In Proc. IUI, 2003

[12] S. Debnath, et al., "Feature Weighting in Content Based Recommendation System Using Social Network Analysis," In Proc. WWW, 2008

[13] R.J. Mooney and L. Roy, "Content-Based Book Recommending Using Learning for Text Categorization," In Proc. SIGIR, 1999.

[14] M. Tsagkias, et al., "News Comments: Exploring, Modeling, and Online Prediction," In Proc. ECIR, 2010.

[15] T. Yano, and N.A. Smith,, "What's Worthy of Comment? Content and Comment Volume in Political Blogs," In Proc. AAAI, 2010.

[16] J. Chen et al., "Short and Tweet: Experiments on Recommending Content from Information Streams," In Proc. CHI, 2010.

[17] S. Faridani, et al., "Opinion Space: A Scalable Tool for Browsing Online Comments," In Proc. CHI, 2010.

DOI: 03.CSS.2011.1.65