Enhanced Information Retrieval Using Domain-Specific Recommender Models

Wei Li Debasis Ganguly Gareth J. F. Jones
Centre for Next Generation Localisation
School of Computing, Dublin City University, Dublin 9, Ireland
{wli, dganguly, gjones}@computing.dcu.ie

Abstract. The objective of an information retrieval (IR) system is to retrieve relevant items which meet a user information need. There is currently significant interest in personalized IR which seeks to improve IR effectiveness by incorporating a model of the user's interests. However, in some situations there may be no opportunity to learn about the interests of a specific user on a certain topic. In our work, we propose an IR approach which combines a recommender algorithm with IR methods to improve retrieval for domains where the system has no opportunity to learn prior information about the user's knowledge of a domain for which they have not previously entered a query. We use search data from other previous users interested in the same topic to build a recommender model for this topic. When a user enters a query on a topic, new to this user, an appropriate recommender model is selected and used to predict a ranking which the user may find interesting based on the behaviour of previous users with similar queries. The recommender output is integrated with a standard IR method in a weighted linear combination to provide a final result for the user. Experiments using the INEX 2009 data collection with a simulated recommender training set show that our approach can improve on a baseline IR system.

Keywords: Domain-Specific Information Retrieval, Recommender Algorithm.

1 Introduction

The ever increasing volume of information available in our daily lives is creating increasing challenges for document retrieval technologies. One area of growing interest in information retrieval (IR) research is the exploration of methods to enable users to find documents which meet their personal information needs by taking advantage of their previous search history. This is the focus of the area of Personalized Information Retrieval (PIR) which seeks to form a model of each user's search interests using their previous search history, and then uses this to assist in more reliably retrieving documents of interest to this user in subsequent search activities. Where the user is searching in a topical area of on-going interest such an approach can prove effective. However, in practice, users may enter queries on new topics which they have not searched on previously. The related field of Recommender Systems (RSs) exploits ratings of items from multiple users to make predictions of items which future users interested in the same topic may find useful. In recent years, RSs have started to appear in many applications where user feedback is available, for example in online applications such as *YouTube*, *Amazon*, and *Ebay*. These systems record the behaviour of users to build interest models of user interests, and use these to predict items which may be of interest to a current user based on feedback from previous ones.

Existing PIR methods require personal information from the specific user in order to build user profiles. This data can be collected by asking users to input their personal information preferences, including for example topics of interest or keywords, recording their search queries and clicking and viewing behaviour when browsing retrieved results or by asking them to rate some items or give other explicit feedback. In other web search personalization technologies, data is collected without user involvement by exploiting the clustering of retrieved documents in order to create a complete personal user profile based on characterization of their search history. These approaches have been found to perform well in the modelled domains [8][9]. However, this approach will not work for new domains where the individual user has not provided personalized information, and it is not realistic to gather such information from them before retrieval operations begin. In this situation it is desirable to make use of any information which is available from previous searchers with similar interests to improve retrieval effectiveness for a user without previous search experiences in the topical domain of their query. To do this, we propose to gather feedback from previous user queries, either recording explicit feedback of relevance of retrieved items to a query or implicit feedback in terms of the time a user dwells on each document. This feedback information then can be used to train recommender models for potentially interesting documents for any new searchers who is interested in this topical domain. In this work we introduce an approach to do this by combining recommender technologies with a standard IR method to produce domain-specific IR where user driven domain models are used to enhance the effectiveness of ranked IR.

We explore our proposed method using a simulated search scenario based on the INEX 2009 Wikipedia document collection. We simulate previous user search behaviour by automatically constructing variations of selected search topics to train a recommender model for the topical domain of each search topic. These recommender models are then used in combination with a language modelling based ranked IR system for search with the original INEX 2009 search topics. The combined search method shows improvement in IR search effectiveness for both precision and recall metrics over a baseline standard ranked IR approach.

The remainder of this paper is organized as follows: Section 2 provides a brief review of relevant existing research in PIR and RSs, Section 3 presents the framework of our proposed combined domain-specific IR model, Section 4 describes our experiments using the INEX 2009 test collection and gives experimental results, and finally Section 5 provides conclusions of our work so far and details of our planned further investigations.

2 Related Work

A number of existing studies have exploring the topic of PIR aim to provide users with more personalized information provision, while other studies have explored the development of RSs. Personalization involves capturing the search interests of individuals and using these to train individual user interest models [25]. There are two broad methods of capturing information for personalization: 1) implicit feedback, where user interests are inferred from their behaviour such as which documents they click on in the output of a search, their reading time for retrieved documents or their scrolling actions on a document; 2) explicit feedback where users manually confirm document relevance or their topical interests [25]. Both IR and RS use these two methods to perform personalization. In this section we look first at existing work in PIR and then review relevant studies looking at RSs.

2.1 Personalized Information Retrieval

PIR is currently being explored mainly in the area of Web search [25]. For example, some standard search engines are examining implicit feedback (mainly by extracting some useful information from the documents which a user has so far viewed) to refine the user's query to provide a more personalized response, e.g. Google, Yahoo! Meanwhile some web search applications are exploring explicit feedback and hybrid approaches combined implicit and explicit feedback, e.g. Flickr, Youtube. These systems ask users to provide tags for source collections or to express their personal descriptions or opinions about some items. For instance, in Flickr, users store and annotate their own photos. These tags can be considered to be expressions of user interests, and can be used to build user profiles which can be exploited in personalized search. Our research is currently looking at only the use of implicit feedback for domain-specific IR, since gathering explicit feedback in the environments that we are considering would be less practical in terms of user participation.

In addition to web search, PIR is also becoming an important factor in other areas, e.g. education, healthcare. Explicit feedback cannot always be gathered in these areas since users may be reluctant to express their opinions or give ratings to items. Because of this, many researchers focus only on collecting implicit feedback. Some studies take account of information about users' behavioural information, such as clicking on information, dwell time when browsing, etc. which can be obtained implicitly from user observation. Kelly and Belkin [11] report that using only display time information averaged over a group of users to predict document usefulness is not likely to be accurate, nor is it accurate using display time for a single user without taking into account contextual factors [15]. For this reason, [15][1] also take account of information about users and their context information beyond their queries. This additional information is often gathered implicitly from user behaviour and contextual data, topic knowledge and task information [15][1]. In our research looking at search in a specific domain, dwell time is the most important factor since it is the only personal data that we can gather from users. In the environment we are working with the topic knowledge of the user and the associated contextual data is unavailable. Thus despite its apparent limitations we are exploring whether dwell time can be exploited as useful for information for the construction of domain-specific recommender models.

2.2 Recommender System

RSs attempt to recommend items that are likely to be of interest to users [25]. Typically, a RS compares user profiles with some reference characteristics, and uses these to predict the rating that a user may give to a new item which he has not considered yet. These characteristics may be associated with the information contained in the item (the content-based approach) or the user's social environment (the collaborative filtering approach) [25]. In this paper we consider only the collaborative filtering approach to RS, other recommender algorithms will be the subject of future work. As exemplified in [22][16], a RS collects user profile information in the same ways as IR systems. Since, as described earlier, we are unable to collect explicit feedback in our environment, we consider collection of implicit feedback. For our investigation, implicit data collection includes:

- Observing the items that a user views.
- Analyzing item/user viewing time.
- Keeping a record of the items that a user has purchased.
- Obtaining a list of items that a user has listened to or watched on their computer.
- Analyzing the user's social network and discovering similar likes and dislikes.

RSs compare the collected data to similar or non-similar data collected from previous users. A list of recommended items can then be calculated for the current user. For our recommender model, we simulate collection of user data from the first two sources: observing the items user views and the viewing time for that item. However, in this preliminary experiment, we assume that when the user inputs a query to our model, we compare the similarity between this query and previous users search information to select a suitable recommender model.

The collaborative filtering approach makes automatic predictions about the interests of a user by collecting preference information from many other users [25]. There are different types of collaborative filtering methods available: memory-based (measures the similarity between pairs of users to give the prediction, the Pearson scheme is a well-known memory-based scheme), model-based (finds patterns from the training data, such as SVD, Bayes methods [25]) and the rating-based approach (predicts how a user would rate a item from other users rating, such as the *inDiscover* website) [13]. In our investigation, we explore the rating-based collaborative filtering approach. In our work we chose the Weighted Slope One algorithm to compute predictions since it is efficient to query, reasonably accurate, and supports both online querying and dynamic updates, which makes it a good candidate for real-world systems [13]. The Weighted Slope One algorithm comprises of the following two steps:

- Looking for other users who rate the same document as the current user.

- Using the ratings from these like-minded users to give predictions for our current user.

Rating-based collaborative filtering requires data pairs from a user: the document ID and its relevance weighting entered either explicitly or inferred from implicit data such as viewing time. The response of a RS to a query pair is an array of recommended pairs (document ID, rating) of documents based on the training data captured from previous users, which the current user has not rated yet.
For simplicity of explanation in our investigation individual users are assumed to enter one query on a topic, but this need not be the case in an operational system.

3 Combining Information Retrieval with Domain-Specific Recommender Models

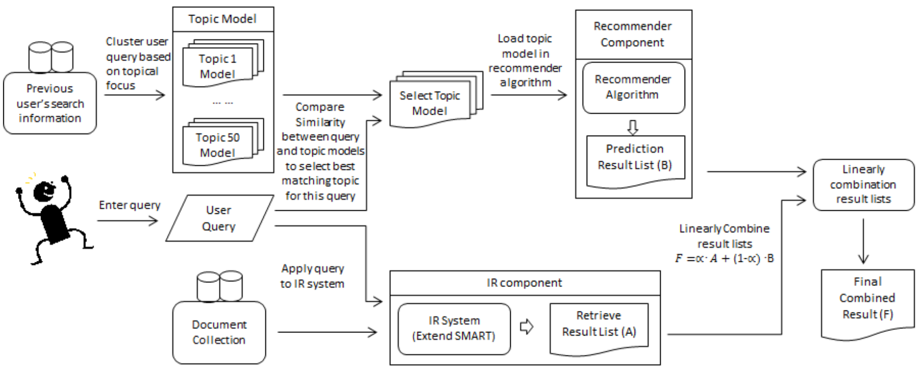

Fig. 1. Overview of the enhanced domain-specific IR incorporating the recommender component.

Our method for integrating domain-specific recommender models with information retrieval proceeds as follows:

- The system records each query entered by previous users to search the available document archive, and implicit feedback from the users of the relevance rating of each retrieved document to viewed by each document, indicated by the time that the searcher spends on viewing the document.
- The ratings of each viewed document for each topical query domain are then used to train a recommender model this domain using the Weighted Slope One algorithm.
- When a query is entered into the combined search method, a standard IR technique is used to retrieve search results from the available document collection. This query is also used to select an appropriate recommender model from the available domain-models generated from previous search data. The RS is then used to give predictions of potentially relevant documents based on the selected recommender model.

- The results of the IR search and RS predications are then integrated using a linear combination of the scores for each retrieved document. Documents are then re-ranked using the combined scores and returned to the user.

Fig.1. shows in the combined domain-specific IR and RS model. This has two components: IR search and recommender prediction. In our experimental implementation of this approach, we use the extend SMART IR system to use a language modelling IR method to retrieve the results for IR component [18].

The training and prediction of each recommender domain-model operates as follows: the domain-model training set for each recommender is based on a set S of all previous queries closely related to the new query. This can be viewed as the following matrix (Equation 1): where $P_{n,m}$ is a pair of data (D_m, $R_{n,m}$), where D_m is document m and $R_{n,m}$ is the rating query to document m for previous query n. This information is then used to run the Weighted Slope One algorithm (2).

$$S = \begin{bmatrix} P_1 \\ P_2 \\ P_3 \\ \dots \\ P_n \end{bmatrix} = \begin{bmatrix} P_{1,1} & P_{1,2} & P_{1,3} & \dots & P_{1,m} \\ P_{2,1} & P_{2,2} & P_{2,3} & \dots & P_{2,m} \\ P_{3,1} & P_{3,2} & P_{3,3} & \dots & P_{3,m} \\ \dots & \dots & \dots & \dots & \dots \\ P_{n,1} & P_{n,2} & P_{n,3} & \dots & P_{n,m} \end{bmatrix} \quad (1)$$

$$P_{(u)_j} = \frac{\sum_{i \in (S-\{j\})} (u_i + dev_{j,i}) \cdot card(S_{j,i})}{\sum_{i \in (S-\{j\})} card(S_{j,i})} \quad (2)$$

This algorithm can take into account both the information from other users who rated the same document and from other documents rated by the current user. For our current experiment we only consider the former. The number of ratings is also observed which means the number of users who rated the pair of items i and j are recorded and considered. If we know that user u gives rating u_i to item i, then we can predict the rating u_j that this user will give to item j based on all previous user information in the recommender set S. This is computed by Equation 2: where $P_{(u)_j}$ is the prediction of the rating that user u will give to item j; $card(S_{j,i})$ is the number of previous queries receiving a rating for i and j in set S; $dev_{j,j}$ is the average deviation of document I with respect to item j, computed using Equation 3.

$$dev_{j,i} = \frac{\sum_{P \in S_{j,i}} P_{k,j} - P_{k,i}}{card(S_{j,i})} (i \neq j) \quad (3)$$

The similarity score of an item j with respect to query q (denoted as $RS(q|j)$), is computed using the standard language modelling approach implemented into the SMART retrieval system (see Equation 4) [4]. Equation 4 ranks a document j by the probability of generating the given query q from it, denoted as $RS(q|j)$. $P(t|j)$ denotes the probability of generating a term t from document j and $P(t)$ denotes the probability of generating it from the collection, λ being the smoothing parameter to account for the inverse document frequency (*idf*) factor of a term t. The final weight of document j (FW_j) is computed using a linear combination of these two scores as shown in Equation 5.

$$RS(q \mid j) = \prod_{t \in q} \lambda \cdot P(t \mid j) + (1-\lambda) \cdot P(t) \tag{4}$$

$$FW_j = \alpha \cdot P_{(u)_j} + (1-\alpha) \cdot RS(q \mid j) \tag{5}$$

For IR systems, the main challenge is to improve the search results for a user's query, in this integrated method, a recommender algorithm is exploited to address this challenge. On the other hand, the key problem for RSs is cold started, the Weighted Slope One algorithm needs other users search information to give predictions for the current user. However, if the recommender set S suffers from data sparse, the prediction results will not be worth considering. In practice the data sparse condition will apply for some user's queries. In this case there will be not be an effective topical domain model available for them. In this case the output of recommender algorithm will be empty or unreliable. Thus integration is only advisable if the recommender domain model has sufficient training data, otherwise conventional IR approach should be preferred. The question of when to apply our integrated approach in the case of limited training data will be considered as part of our future work.

4 Experiment

To investigate our proposed approach to domain-specific search, we require a suitable set of experimental data which enables us to build recommender models for a range of topical search interests based on previous users interaction behaviour within a suitably challenging IR task. These requirements mean that experimentation poses many challenges. Since this is a new research area, there are no suitable test collections readily available. Ideally we would have access to real collections and importantly large logs of queries and interaction data from real users querying this data. However, since we do not have access to this type of datasets, and it is unrealistic for us to be able to collect such a dataset working in an academic environment or to gain access to such data from other sources, we must seek ways to simulate them in order to explore our proposed approach. In order to conduct this initial study we extended an existing test collection to simulate our search environment. We assume a scenario of a visitor to a museum with an interest in a topic entering a query to identify items which may be of interest to them within the available collection. Subsequently other users enter similar but different queries on the same topical area. The relevant documents returned for each of these queries can then be gathered to form the training set for a recommender model for this topic. The relevance rating for each occurrence of each relevant document is taken from the viewing time for each relevant document. Thus a document retrieved for many queries with high ratings will be given a high relevance prediction value by the RS for this topic. For our experiment we assume search interests for documents on a number of separate topical areas and use these to build recommender model for each of them.

The following subsections describe the development of the test collection used for our initial investigation.

4.1 Data Collection

The INEX 2009 Wikipedia document collection was selected as our starting data collection. For this investigation we simulated previous user search interaction information as follows: 20 topics were chosen from the INEX 2009 topic dataset, the criteria for choosing topic is that they should be 4 words or longer in length. 10 variations of each topic were created as a simulation of similar queries in the same topical area entered by previous users. Topic variants were made by randomly deleting one or two words from the original topic, hence the need for topics statements of 4 or more words. For example, for the original topic: *"Physicists scientists alchemists periodic table elements",* two of the variations were as follows: 1) *Physicists scientists alchemists periodic results; 2) Physicists scientists alchemists table results.*

This is obviously a very simple strategy for creating topic variations, but serves to enable us to carry out our current experiment. A particular issue which needs to be considered when modifying queries in this way is the potential impact on the set of documents which are relevant to each topic. For this initial investigation, the relevance set is assumed to not vary a lot for each topic variation. We are currently working on more sophisticated methods to simulate query variants to obtain more realistic training datasets for our experiments.

In this experiment, each recommender model is built in the following way:

- For each original topic, make 10 variants by random deleting one or two words.
- The 10 topic variants for each search topic were entered as queries into the extended SMART system to obtain 10 ranked lists of potentially relevant documents.
- The retrieved results for the 10 variations of each topic were clustered in one group. As described above, the aim of this step is to simulate results obtained for 10 different users who interested in the same topic. The topic variations mean that slightly different result lists are obtained for each pseudo user search query.
- The 10 ranked lists for each topical area were compared against the qrel files of the original topics to identify retrieved true relevant documents retrieved for each topic variant. Rating values were then assigned to each document randomly. The ratings simulated browsing time as an indication of implicit feedback. These were assigned in the range 0.5-1.0 for each relevant document for the original topic, and 0.0-0.49 for documents which were non-relevant.

- Each of the top 150 documents in the retrieved ranked lists with rating information is seen as one previous users searching behaviours. The processed retrieved ranked for the 10 variants were integrated into one group and used as a recommender model for their corresponding original query. We thus obtained our simulation data for previous users searching the document collection.

4.2 Experiment Setup

Recommender models for the 20 selected topics were built as described in section 4.1. For our experiment we assume that a searcher has an interest in one of our 20 topical domains and enters the original search query to look for relevant documents that they might be interested in. The query is applied to the extended SMART retrieval system to obtain a ranked document list. This represents our baseline retrieval output. The ranked IR retrieval list is then compared to find the appropriate recommender model from those available. The recommender model selection proceeds as follows. We assume the retrieved ranked list is a vector $Q=(d_{1,q}, d_{2,q}, d_{3,q},\ldots\ldots,d_{t,q})$,we have 20 recommender models in our experiment $(R_1, R_2, R_{3,\ldots\ldots}, R_{20})$, recommender model k is a set S_k (Equation (1)) and can also be viewed as a vector, i.e. recommender model k can be seen as a vector $R_k = (P_{1,k}, P_{2,k}, P_{3,k,\ldots\ldots},P_{n,k})$ $(j\in[1,20])$, where $P_{i,j}$, is the result for one previous query in recommender model k. The similarity between the query vector and each recommender vector is:

$$Sim(Q,R_k) = \sum_{t=1}^{20} f_{d_{j,q},R_k} \tag{6}$$

where $f_{d,R}$ is the frequency of item d in the recommender model R_k based on set S_j. The recommender model with the highest similarity is selected as the best matching topical domain for the input query. Here t from 1 to 20, which means we only go through the top 20 documents in the retrieved ranked list. This is then used to calculate the prediction of the rating that our current user would give to each of the available documents. Finally, the recommended ranking results are linearly combined with the baseline retrieval list to output our final integrated results. In this experiment, the parameter $\alpha = 0.25$ in Equation (5) was set using informal empirical experimentation.

4.3 Result

Our scenario task is a user entering a query on entering a museum. We thus focus on providing results to users which show them where they should go next. Our aim here then is to find the most relevant documents on the top of the list in order to give the user most precisely results which direct them to really interesting documents in their next step, i.e. we are interested particularly in precision at high rank cutoff of the retrieved list.

Results for this experiment are calculated using the standard trec_eval software, and are shown in Table 1. Our baseline (BL) IR results are output by the SMART system extended using language modeling IR. The domain-specific IR (DSIR) results combine this result with the recommender system as shown in Equation (5). Results are shown for total number of documents retrieved, no of relevant items retrieved, precision at rank cutoffs of 5, 10, 20 and 100, and standard Mean Average Precision (MAP).

Table 1. Retrieval results for 20 topics with simulated recommender training.

Topic	20_Topic_BL	20_Topic_DSIR
Total Number Retrieved	29950	29950
Total Number Relevant	1166	1166
Total Relevant Retrieved	355	355
MAP	0.0744	0.1303 (+75.03%)
P@5	0.2600	0.5000 (+92.30%)
P@10	0.2200	0.3560 (+61.81%)
P@20	0.1750	0.2000 (+14.23%)
P@100	0.0835	0.1320 (+58.08%)

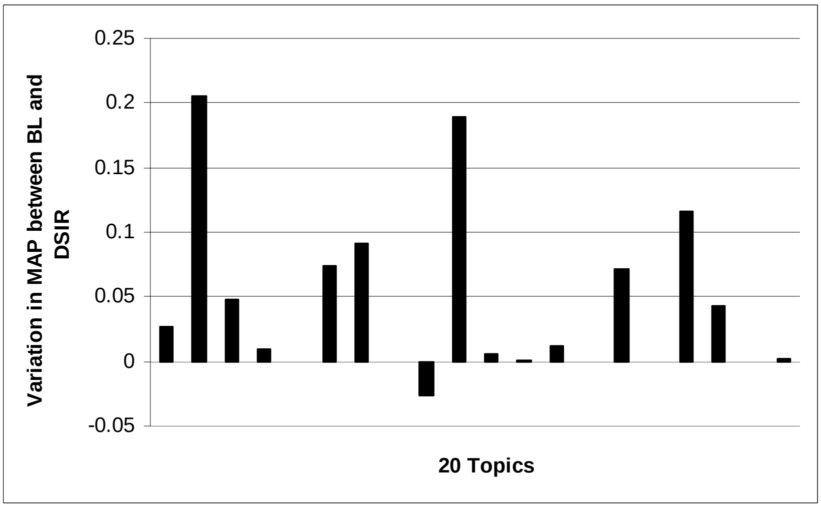

Fig. 2. Variation of MAP between BL and DSIR approaches for 20 original test topics. Calculated as MAP of DSIR - MAP of BL.

From Table 1 we can see that the DSIR approach achieves a MAP of 0.1303 which represents an impressive increase of +75.03% on the baseline IR system. The precision at top 5 cut-off is increased from baseline 0.2600 to 0.5000 (+92.30%). This is partially matches our aim of seeking to promote relevant documents to the top of the ranked list. This demonstrates that the recommender algorithm can help to aid standard IR methods. Figure 2 shows the deviation of MAP between BL and DSIR approach for 20 original topics, calculate by the MAP of DSIR - MAP of BL. Form Figure 2, we can clearly see that for the selected 20 topics, the average performance of DSIR is better than our baseline. The reason that it cannot perform well on all topics is that its results depend on the previous users visiting information. If the recommender model we choose for the current user is correct and contains items that are relevant to its topic, the recommender algorithm will locate it and give it as a prediction to the user. In this experiment, of the 20 evaluation topics, 4 of them were assigned to the wrong recommender model. Improving the reliability of recommender assignment will be an area of our further work.

5 Conclusion and Future Work

In this paper we have proposed a domain-specific IR method combining ranked IR and RSs methods. Experiments with a simulated search environment show that this integration has the potential to improve retrieved results over standard IR methods. While this initial experiment shows promising results, further work needs to be done to develop a more realistic experimental environment. For example, to use a more sophisticated model for query variations in training the RSs. Additionally, while the results so far are encouraging, there are various ways to improve the baseline IR, including methods such as relevance feedback. In our further work, we will explore integration of methods such as these to compare their contribution to improving IR with that of the recommender based approach.

The scenario we are exploring here considers a searcher exploring a new domain of interest. Thus we expect our searcher to be exploring a number of items. When doing this we can make use of feedback as they explore items to adapt the RS in a personalized manner. Also, if they are learning about a new topic, there will often be a preferable order in which information should be viewed. So ultimately we are interested not just in identifying relevant items, but also determining the order in which they are presented in order to maximize the efficiency with which information is provided to the searcher. New evaluation strategies will be required in order to measure the effectiveness with which relevant items can recommended to the searcher in an optimally efficient sequence.

6 Acknowledgements

This research is supported by the Science Foundation Ireland (Grant 07/CE/I1142) as part of the Centre for Next Generation Localisation (CNGL) project at Dublin City University.

References

1. Belkin. N. J. Some (what) Grand Challenges for Information Retrieval. *ACM SIGIR Forum* 42(1):47-54. (2008)
2. Billsus D. and M. Pazzani. Learning Collaborative Information Filtering. *In Proceedings of the AAAI Workshop on Recommender Systems* (1998)
3. Campi A., M. Mazuran and S. Ronchi, Domain Level Personalization Technique. *In Proceedings of VLDB 2009*. (2009)

4. Chee S. H. S., J. H. Wang and K. Rectree, An Efficient Collaborative Filtering Method. *In Proceedings of the 3rd International Conference on Data Warehousing and Knowledge Discovery*, pages 141-151. (2001)
5. Geva S., J. Kamps, M. Lethonen, R. Schenkel, J. A. Thom and A. Trotman. Overview of the INEX 2009 Ad Hoc Track. *In Proceeding of INEX 2009*, pages 4-25. (2009)
6. Herlocker J., J. Konstan, A. Borchers, and J. Riedl. An Algorithmic Framework for Performing Collaborative Filtering. *In Proceedings of ACM SIGIR 1999*. (1999)
7. Hiemstra D. Using Language Models for Information Retrieval. PhD thesis, Center of Telematics and Information Technology, University of Twente. (2001)
8. Jansen B. J., A. Spink, J. Bateman, and T. Saracevic, Real Life Information Retrieval: A Study of User Queries on the Web. *ACM SIGIR Forum* , 32(1). (1998)
9. Jeon H., T. Kim and J. Choi. Adaptive User Profiling for Personalized Information Retrieval. *In Proceedings of ICCIT 2008*, pages 836-841 (2008)
10. Jin R. and L. Si. A Study of Methods for Normalizing User Ratings in Collaborative Filtering. *In Proceedings of ACM SIGIR 2004*, pages 568-569. (2004)
11. Kelly D. and N. J. Belkin. Display Time as Implicit Feedback: Understanding Task Effects. *In Proceedings of SIGIR 2004*, pages 377-384. (2004)
12. Kuhlthau C. C. Inside the Search Process: Information Seeking from the User's Perspective. *Journal of the American Society for Information Science*. 42(5):361-371. (1991)
13. Lemire D. and A. Maclachlan. Slope One Predictors for Online Rating-Based Collaborative Filtering. *In Proceedings of SIAM Data, SDM'05* (2005)
14. Linden G., D. Jacobi, A. Jennifer and E. A. Benson. Collaborative Recommendations Using Item-to-item Similarity Mappings*. In Proceedings of SPSS.* (2001)
15. Liu J., and N. J. Belkin, Personalizing Information Retrieval for Multi-Session Tasks. The Roles of Task Stage and Task Type. *In Proceedings of ACM SIGIR 2010*. (2010)
16. Oard D. and J. Kim. Implicit Feedback for Recommender System. *In Proceedings of the AAAI Workshop on Recommender Systems.* (1998)
17. Pennock D. M. and E. Hirvutz. Collaborative filtering by personality diagnosis. *In Proceedings of IJCAI-99*. (1999)
18. Ponte J. M. and W. B. Croft. A Language Modeling Approach to Information Retrieval. *In Proceedings of ACM SIGIR 1998.* (1998)
19. Saric A., M. Hadzikadic and D. Wilson. Alternative Formulas for Rating Prediction Using Collaborative Filtering*. In Proceedings of ISMIS 2009*. (2009)
20. Sarwar B. M., G. Karypis, J. A. Konstan and J. Riedl. Item-Based Collaborative Filtering Recommender Algorithms. *In Proceedings of WWW10*, pages 285-295 (2001)
21. Shaw J. A. and E. A. Fox, Combination of Multiple Searches. *In Proceedings of ACM SIGIR 1993*. (1993)
22. Song Y., N. Nauyen, L. He, S. Imig, and R. Rounthwaite. Searchable Web Sites Recommendations. *In Proceedings of the Fourth ACM international conference on Web Search and Data Mining (WSDM 2011*). (2011)
23. Vallet D., I. Cantador, and J. M. Jose. Personalizing Web Search with Folksonomy-based User and Document Profiles. *In Proceedings of ECIR 2010*, pages 420-431 (2010)
24. White R. and D. Kelly. A Study of the Effects of Personalization and Task Information on Implicit Feedback Performance. *In Proceedings of CIKM 2006*, pages 297-306. (2006)
25. Wikipedia, http://en.wikipedia.org