Utilizing Recommender Algorithms for Enhanced Information Retrieval

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

ABSTRACT

Retrieving relevant items which meet a user's information need is the key objective of information retrieval (IR). Current IR systems generally seek to satisfy search queries independently without considering search history information from other searchers. By contrast, algorithms used in recommender systems (RSs) are designed to predict the future popularity of an item by aggregating ratings of the reactions of previous users of an item. This observation motivates us to explore the application of RS methods in IR to increase search effectiveness. In this study, we examine the suitability of recommender algorithms (RAs) for use in IR applications and methods for combining RAs into IR systems by fusing their respective outputs. A novel RA is proposed to enhance the RS performance in our integrated application. Experimental results are reported for an extended version of the FIRE 2011 personalized IR data collection. Noticeably better results are obtained using our approach.

Categories and Subject Descriptors: H.3.3 **[Information Search and Retrieval]**: Retrieval Models

Keywords: Recommender Algorithm, Information Retrieval

1. INTRODUCTION

The increasing volume of online information is creating ever greater challenges for people to identify personally useful information. Personalized information retrieval (PIR) is used to address this problem by collecting personal information about a user to build a user profile. PIR seeks to use this profile to give retrieval results which better meet the informational needs of the individual user than those of a standard information retrieval (IR) system. However, in many situations there may be no opportunity to learn about a user's specific interests or knowledge in relation to a particular search query if they have not previously entered queries on this topic. To improve search in such situations, the experiences and behaviour of other searchers who have previously entered similar queries could be used to build a model of user behavior in this topical category. One method to achieve this is via recommender systems (RSs) which use historical user ratings of related items to predict items to a current query. RSs incorporated into a search engine can potentially enhance search effectiveness by combining a recommendation component into the IR system.

Our earlier work [1] concentrated on a single recommender algorithm (RA). In this study, we examine the performance of 5 different RAs in our IR model, results from these experiments show that these existing algorithms, while effective in standard RSs, have weaknesses in our IR application. We describe and evaluate a novel extended RA designed to address these problems for our application. We further exploit the hybrid recommender approach to address the cold start and sparse data problems of RAs. Finally we explore use of alternative fusion methods for combination of the standard IR and RS components. Experiments are conducted on an extended version of the test collection developed for the FIRE 2011 PIR task [6]. The remainder of this paper is organized as follows: Section 2 overviews the existing recommender schemes examined in this study, Section 3 introduces our proposed RA, Section 4 describes our integrated IR model, Section 5 reports our evaluation of the model, and finally Section 6 gives conclusions of our work so far and plans for our further investigations.

2. RECOMMENDER TECHNIQUES

The goal of a RS is to generate meaningful recommendations for users of items which might interest them. This section briefly overviews the relevant details of three RS methods.

2.1 Content-Based Filtering Algorithm

The first approaches to information filtering were based on the content of each item [5]. These RSs provide recommendations by comparing representations of the content of an item to a representation of the user's interests. While effective in some situations where items and user interests can be sufficiently represented by a set of keywords, these content-based techniques have several limitations [3]:

1. Content limitation - the content of documents should be machine indexable. For example, multimedia content may lead to problems in carrying out content analysis.
2. Inability to evaluate the quality of an item.
3. No way of finding serendipitous items which are interesting for other users.

2.2 Collaborative Filtering Algorithm

Collaborative filtering (CF) systems [11] recommend items which have received high ratings from other users' queries which are similar to the current query. However, they are sensitive to common problems of RSs. The two most important problems are:

- Data sparsity problem [9]: Usually, each user only rates a small number of available items for each query. In such cases, it is a challenge to identify similarities among different queries or items.
- Cold start problem [10]: User-side cold start problem occurs where users have not rated enough items. This leads to RSs that are unable to determine their preferences. This problem also affects new items as they will not be recommended until enough users have rated them.

The following subsections overview a number of different CF algorithms found in the literature that we later evaluate and

OAIR'13, May 22-24, 2013, Lisbon, Portugal.

compare for use in our RS enhanced IR model.

Item-based CF

Item-based CF algorithms look for items similar to each item rated for the current query. They follow two steps: calculate the similarity between each item pair; compute the prediction using a similarity matrix based on the current query's rating information. In our study, the adjusted cosine similarity [9], which utilizes rating information to compute similarity between items, is used to build the similarity matrix. We sum the ratings that the current query has given to items by using these items' similarity with other items to compute the prediction.

Cluster-based smoothing

Cluster-based smoothing groups the users into clusters to satisfy two objectives: to increase the density of the rating matrix, approaching unknown ratings with the cluster mean, and to increase the scalability. The details of this algorithm can be found in [13]. In our work, we use this method as follows:

1) The cluster mean is used to fill user profiles.
2) Select the closest topic category to the current query.
3) If an item occurs in the selected topic category, its prediction weight is computed using the weighted mean in this topic category.

Tendency-based CF

Cacheda [3] proposes a tendency-based CF algorithm, instead of looking for the similarity relations between items, this looks at the differences between items. They suggest that users rate items in different ways, these variations are related to their differences in opinions and interests. Besides this, they also observe that even when users have similar preferences, they might rate items in different ways. Some users are more inclined to give positive ratings, leaving negative ratings for really bad items; others might save their highest ratings for the best items and tend to give negative ratings to all other items. However, these variant rating strategies are an accurate indicator of the quality of each particular item and its utility for the users. These variant rating approaches are interpreted in two ways in [3]: users' tendencies and item tendencies. Users' tendencies refer to whether a user tends to rate items positively or negatively. Item tendency is defined as whether users consider this item an especially good or a bad item. The results of experiments in [3] show that this tendency-based RA has high computational efficiency.

Rating-based CF

Rating-based CF algorithms use the users' rating information to compute prediction of items likely to be of interest. The weighted slope one (WS1) algorithm is chosen in our work, since was shown to be highly effective in [7]. WS1 takes into account information from both other users who rated the same items and from the other items rated for the current query, and also considers the number of ratings. This scheme assumes that the ratings from a given user are represented as an incomplete array *u*, u_i is the rating of this user given to item *i*. The subset of items consisting of all items which are rated in *u* is *S(u)*. The number of elements in a set *S* is *card(S)*. The WS1 procedure is: for a training set *X*, two items *j* and *i* with ratings u_j and u_i respectively in an array *u* ($u \in S_{j,i}(X)$), consider the average deviation of item *i* with respect to item *j*, as shown in Equation (1), the prediction for item *j* is defined in Equation (2).:

$$dev_{j,i} = \sum_{u \in S_{j,i}(X)} \frac{u_j - u_i}{card(S_{j,i}(X))} (i \neq j) \qquad (1)$$

$$P^{WS1}(u)_j = \frac{\sum_{i \in S(u)-\{j\}} (u_i + dev_{j,i}) \cdot card(S_{j,i}(X))}{\sum_{i \in S(u)-\{j\}} card(S_{j,i}(X))} \qquad (2)$$

2.3 Hybrid Approach

Hybrid recommendation methods combine both content-based and collaborative approaches [8][2]. In order to leverage the strengths of both content-based and collaborative methods, several hybrid approaches have been proposed which combine these two techniques. One simple approach is to allow both methods to produce recommendations separately, and then to merge their results to generate the final results [8]. We employ this approach in the last step of our integrated IR model.

3. PROPOSED RECOMMENDER ALGORITHM

From the results of our initial study [1], we observed that the WS1 scheme relies on ratings information from users with similar interests. On inspection it can be seen that a problematic situation can arise in this case. Consider Equation (2), if a user rates an item *A* as 0.2, and 20 users rate both item *A* and *B* with resulting average deviation between them of 0.6, while 2,000 users rate both item *A* and *C* with average deviation of 0.55, then item *B* is preferred (0.8 vs 0.75). However, for an IR system, if an item has been rated by the majority of users for a particular query, and the average rating for this item is higher than average value, this item must have the potential to be relevant to this query. Based on this hypothesis, we anticipate item *C* has the potential to be more relevant to item *A* than item *B*. To address this problem, we count the frequency of each item and computer the average rating for each item, and use these factors to better reflect desirable IR behavior. The relationship between these factors is shown in Table 1, for item *i*, where F_i denotes its frequency, $\overline{R_i}$ is its average ratings, relevance assessment means the relevance between item *i* and the current query.

Table 1. The relation between F_i, $\overline{R_i}$ and relevance

	F_i	$\overline{R_i}$	Relevance Assessment
1	High	High	High Relevance
2	High	Low	Less Relevance
2	Low	High	Less Relevance
3	Low	Low	Non Relevance

Based on this observation, we propose a novel extension to the WS1 which we refer to as the *popularity focused weighted slope one (PWS1),* shown in Equation (3). This algorithm uses the relationship presented in Table 1. to extend the existing WS1 algorithm to make it more effective for our IR model.

$$P^{PWS1}(u)_j = P^{WS1}(u)_j \cdot [(\lg F_j + k)/(1-(\overline{R}_j/(1+\overline{R}_j)))] \qquad \mathbf{(3)}$$

where *k* is a constant value which is used for the condition that when this predicted item frequency is only 1 in a topic category will lead to a zero value for the final prediction for this item. We utilize $1-(\overline{R_j}/(1+\overline{R_j}))$ and $\lg F_j + k$ to control the contribution of item frequency and average rating of this item in the final prediction score. We hypothesize that this novel algorithm will perform better than the other algorithms introduced in Section 2 in improving IR output. In this study, *k=2* was set empirically.

4. INTEGRATED RETRIEVAL MODEL

Our enhanced IR model integrates RS outputs with a ranked IR output. As shown in Figure.1, the IR and RS components operate separately, and are combined in the last stage. Our initial results of this method were very encouraging [1], evaluation was based on simulation of topically related queries, and a single RA and simple fusion strategy were investigated. Our current study extends this to a comparison of multiple RAs and a set of topically related queries gathered from human participants and corresponding manual relevance assessments.

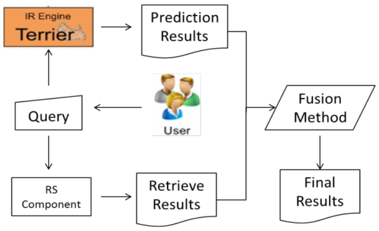

Figure 1. Framework of the integrated Retrieval model.

5. EXPERIMENTAL INVESTIGATION

In this section we describe the data collection and setup for our experiments and give the results and analysis of our study.

5.1 Data Collection

In order to evaluate the effectiveness of our integrated IR model we used the data collected for the FIRE 2011 PIR task [6]. This dataset is based on the FIRE 2011 English ad-hoc document collection composed of news articles from the Indian newspaper *The Telegraph* from 2001 to 2010 and news from Bangladesh. The following steps were used to create user behavior data:

- A number of participant volunteers selected news topics they are interested in, and created a query describing a specific information need within each of their chosen topics. Each was submitted to an IR system (Terrier) to obtain a ranked document list.
- The details of participant's activities were recorded. These included: their name, topic selected, their query, articles viewed, and dwell time viewing on each article. Since news articles are relatively short, we regard the dwell time as a reasonable measure of expected document relevance.
- Each participant was also asked to read the top 30 articles in the ranked list returned for their query, and mark the relevant ones. Note that this relevance assessment collection process is separate with the search log collecting procedure.

Finally, 26 participants contributed 150 TREC formatted queries for the 27 topical categories. Since the participants were given free choice of topics, the queries were distributed unevenly over the available topical categories. 27 test topics were extracted (one at random for each topic category), the remaining 123 queries were used to train the retrieval model. All parameters in this study were trained empirically using this training set.

5.2 Experiment Setup

5.2.1 IR Component

The Terrier BM25 retrieval model was used to generate ranked lists for the IR component. A stopword list of 500 words was used with a Porter stemmer to preprocess the input text. 1000 news articles were returned in the ranked list for each query. For this study, the BM25 parameters were set: $k_1 = 1.2$ and $b=0.75$..

5.2.2 Recommender Component

Content-based filtering (CBF)

For the CBF method, the similarity between the current query and each document needs to be computed. However, the length of a user query is usually too short to compute the similarity to produce a meaningful recommendation. In order to address this problem, we applied the query to the Terrier system using the BM25 retrieval model to obtain a ranked list for this query. Then the top *N* documents in this ranked list were used to generate a centroid document representation for current query (C_q) [4]. In this study, N=*5* was set empirically based on the training set. The adjusted cosine similarity [9] was used to compute the distance between C_q with every document.

Categorizing

The purpose of this step is to attempt to identify the correct topic category for each test query. The categorizing process is:

- Compute document frequency for each document in each topic category. Use top 5 highest frequency documents to generate the centroid document [4] for each topic category.
- Match the query representation (query centroid document) to each topic category by using adjusted cosine function [9]. The closest topic category is selected.

Here top 5 was chosen empirically. The reason for failing to choose the correct topic category on some occasion is that we simply use the 5 highest frequency documents in every topic category to generate its centroid document. Sometimes, too many noisy documents are present in each topic category, such as non-relevant document at high rank, most users view it but with low dwell time. In this case, use only the highest frequency documents to build centroid may lead to topic drift.

Current User Information

Some RSs algorithms introduced in Section 2.2 require rating information for the current query to compute predictions. Our experiment suffers user-side cold start problem. To address this problem, similar to pseudo relevance feedback (PRF) in IR, we apply the query to the Terrier system utilizing BM25 retrieval model to obtain a ranked list. The top *N* documents and their relevance rating obtained from the retrieval results are employed as ratings for the current query. Here *N* was set equal to 5. Alternative values of *N* were investigated. N=*3* contained few documents and may lead to a problem that these 3 documents do not exist in the selected topic category, in which case no recommendation would be given for rating-based and document-based CF techniques. By contrast, N=*10* introduced noise which meant that non-relevant documents were present with high scores, and recommendations tended to be unreliable.

Combination

Four data fusion operators were investigated in this study to test their effectiveness in our integrated IR model: CombSUM, CombMAX, CombMIN and CombMNZ. These fusion methods are widely used for combining the outputs of different indexing schemes [12]. The CombSUM operator is simply the sum of the retrieval status values. The CombMAX/CombMIN operator combined score is the maximum/minimum retrieval status value achieved. CombMNZ is computed by multiplying CombSUM by the number of lists containing this document.

Collaborative filtering

For the other 5 recommender techniques introduced in section 2.2, the experimental setup contained similar 4 steps:

1. Assign the current user query to a topic category
2. Use the selected topic category to compute prediction
3. Apply the query to the Terrier search engine using the BM25 retrieval model to obtain search results
4. Combine the prediction results with IR results

5.3 Experimental Results

Retrieval effectiveness was evaluated using Mean Average Precision (MAP) and precision at cut-off rank. Two baselines were used in order to assess the effectiveness: the standard IR result ranked using the Terrier BM25 retrieval model with PRF query expansion by adding 5 terms from top 5 documents. These numbers were again chosen empirically. Experimental results shown in Tables 2, 3 and 4 for the following runs: Baseline (BL), Baseline with query expansion (BL+QE), Content-based filtering (CBF), Cluster-based CF (CBCF), Tendency-based CF (TBCF), Item-based CF (IBCF), Rating-based CF (IBCF), popularity-focused rating-based CF (PFRBCF).

Table 2 shows the MAP value for the two baselines and the 6 recommender techniques introduced in Sections 2.1 and 2.2 combined with standard IR output. The presented results show that the performance of the CBF method combined with the baseline IR run slightly improves on the baselines, and that our proposed PFRBCF algorithm obtains the best results, which increases MAP by 65.49% compared to the BL run, and by 42.57% compared to BL+QE when using the combMNZ method. Table 3 shows the MAP results for combination of three lists: BL, CBF and one of the other 5 CF technique results. These combined lists indicate that our PFRBCF algorithm outperforms the other runs. The good results presented in Table 2 and 3 reveal that, although RSs share fundamental features with IR, combining IR with RAs can improve IR result rank. Since their goals are not exactly same, standard RAs are not suitable for direct use in this integrated model. Adapting RAs for use in an IR system can make them perform better in our integrated IR model. Since we have 150 queries unevenly distributed across the 27 topic in the dataset, some topic categories suffer a data sparse problem (introduced in Section 2.2). In this case, hybrid recommender approach which incorporates CBF with CF can help our integrated model to address this problem. From Table 3, we conclude that employing the integrated hybrid recommender approach with IR outperforms integrated single RA with IR output. Tables 2 and 3 also indicate that CombMNZ performs best among the four fusion methods examined. MAP and precision at different cut off ranks for the hybrid approach are shown in Table 4. These results show that our methods achieve the greatest improvement for P@5, P@10 and P@20.

Table 2. MAP of 8 different runs using 4 fusion methods.

	comSUM	comMAX	comMIN	ComMNZ
BL	0.1275	0.1275	0.1275	0.1275
BL+QE	0.1480 (+16.1%)	0.1480 (+16.1%)	0.1480 (+16.1%)	0.1480 (+16.1%)
BL+CBF	0.1558 (+22.2%)	0.1550 (+21.6%)	0.1540 (+20.8%)	0.1620 (+27.1%)
BL+CBCF	0.1995 (+56.5%)	**0.2000** (+56.9%)	0.1470 (+15.3%)	0.2006 (+57.3%)
BL+IBCF	0.1705 (+33.7%)	0.1535 (+20.4%)	0.1550 (+21.6%)	0.1890 (+48.2%)
BL+TBCF	0.1995 (+56.5%)	0.1882 (+47.6%)	0.1455 (+14.1%)	0.1862 (+46.0%)
BL+RBCF	0.1768 (+38.7%)	0.1550 (+21.6%)	0.1560 (+22.4%)	0.1950 (+52.9%)
BL+PFRBCF	**0.2011** (+57.7%)	0.1960 (+53.7%)	**0.1990** (+56.1%)	**0.2110** (+65.5%)

Table 3. MAP of 7 runs, Baseline with content-based CF and one of other 5 CF methods in 4 fusion methods.

	comSUM	comMAX	comMIN	ComMNZ
BL	0.1275	0.1275	0.1275	0.1275
BL+QE	0.1480 (+16.1%)	0.1480 (+16.1%)	0.1480 (+16.1%)	0.1480 (+16.1%)
BL+CBCF+CBF	0.2053 (+61.0%)	**0.2157** (+69.2%)	0.1695 (+32.9%)	0.2057 (+61.3%)
BL+IBCF+CBF	0.1935 (+51.8%)	0.1763 (+38.3%)	0.1725 (+37.4%)	0.1990 (+56.1%)
BL+TBCF+CBF	0.2010 (+57.6%)	0.1993 (+56.3%)	0.1648 (+32.1%)	0.2017 (+58.2%)
BL+RBCF+CBF	0.1987 (+55.8%)	0.1765 (+38.4%)	0.1756 (+37.7%)	0.2150 (+68.6%)
BL+PFRBCF+CBF	**0.2231** (+75.0%)	0.2108 (+65.3%)	**0.2095** (+64.3%)	**0.2350** (+84.3%)

Table 4. GMAP, P@5, P@10 and P@20 value for 7 runs on hybrid approach, merged by combMNZ fusion method.

	GMAP	P@5	P@10	P@20
BL	0.0039	0.1173	0.0967	0.0680
BL+QE	0.0039 (+0.00%)	0.1360 (+15.9%)	0.0890 (-7.96%)	0.0710 (+4.41%)
BL+CBCF+CBF	0.0040 (+2.56%)	0.1842 (+57.0%)	0.1400 (+44.8%)	0.0987 (+45.1%)
BL+IBCF+CBF	0.0047 (+20.5%)	0.1627 (+38.7%)	0.1340 (+38.6%)	0.0890 (+30.9%)
BL+TBCF+CBF	0.0042 (+7.69%)	0.1713 (+46.0%)	0.1373 (+41.9%)	0.0933 (+37.2%)
BL+RBCF+CBF	0.0050 (+28.2%)	0.1687 (+43.8%)	0.1343 (+38.9%)	0.0917 (+34.9%)
BL+PFRBCF+CBF	**0.0054** (+38.5%)	**0.1865** (+58.9%)	**0.1520** (+57.2%)	**0.1080** (+58.8%)

6. CONLUSION AND FUTURE WORK

This paper has demonstrated that a recommender component can be successfully applied to improve IR search results for a query by utilizing the search history of previous users. Several different recommender algorithms were investigated with results indicating that standard recommender techniques can improve pure IR system results after integration, but also that adapting rating-based recommender technique using IR features can make them more effective in our integrated IR model. As future work we plan to address the following: examine alternative fusion methods to combine multiple scheme results; investigate how to detect the condition where a query does not belong to one of the topic models, explore how to build a new topic category for queries falling outside the scope of the existing topics; further evaluate the model on a larger dataset to further explore its behaviour and effectiveness.

7. ACKNOWLEDGEMENTS

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

8. REFERENCES

[1] Wei L. and Gareth G.J.F..: Enhanced Information Retrieval Using Domain-Specific Recommender Model. *In ICTIR'11*. (2011)

[2] Basilico J., and Hofmann T.: Unifying Collaborative and Content-Based Filtering. *In ICML'04*. (2004)

[3] Cacheda F., Carneiro V., Fernandez D. and Formoso V.: Comparison of Collaborative Filtering Algorithms: Limitations of Current Techniques and Proposals for Scalable, High-Performance Recommender Systems. *Journal of ACM Transactions on Web (TWEB),* 5.(2011)

[4] Eui-Hone H. and George K.: Centroid-Based Item Classification: Analysis & Experimental Results. *In PKDD'00*. (2000)

[5] Foltz P.W. and Dumais S.T.: Personalized Information Delivery: *An Analysis of Information Filtering Methods*. ACM 35. (1992)

[6] Ganguly D., Leveling J., Keith C. and Jones G.J.F.: Overview of the Personalized and Collaborative Information Retrieval (PIR) Track at FIRE-2011. *In FIRE'11 Worksho*p. (2011)

[7] Lemire D. and Maclachlan A.: Slope One Predictors for Online Rating-Based Collaborative Filtering. *In SIAM* (2005).

[8] Melville P. Mooney R. and Nagarajan R.: Content-Boosted Collaborative Filtering. *In the ACM SIGIR Workshop on Recommender System* (2001)

[9] Sarwar B., Karypis G., Konstan J. and Reidl J.: Item-Based Collaborative Filtering Recommendation Algorithms. *In WWW'01*, China (2001)

[10] Schein A.I., Popescul A., Ungar L.H., and Pennock D.M.: Methods and Metrics for Cold-Start Recommendations. *In ACM SIGIR*. (2002)

[11] Shardanand U. and Maes P.: Social Infomration Filtering: Algorithms for automating "Word of Mouth". *In the SIGCHI (CHI'95)*. Denver, Colorado, USA (1995)

[12] Shaw J.A. and Fox E.A.: Combination of Multiple Searches. *In Second Text Retrieval Conference (TREC-2)*. (1994)

[13] Xue G.R., Lin C.,Yang Q., Xi W., Zeng H.J., Yu Y. and Chen Z.: Scalable Collaborative Filtering Using Cluster-Based Smoothing. *In ACM SIGIR*. (2005)