Enhanced Information Retrieval by Exploiting Recommender Techniques in Cluster-Based Link Analysis

Wei Li
Centre for Next Generation Localisation
School of Computing
Dublin City University, Dublin 9, Ireland
wli@computing.dcu.ie

Gareth G.F. Jones
Centre for Next Generation Localisation
School of Computing
Dublin City University, Dublin 9, Ireland
gjones@computing.dcu.ie

ABSTRACT

Inspired by the use of PageRank algorithms in document ranking, we develop and evaluate a cluster-based PageRank algorithm to re-rank information retrieval (IR) output with the objective of improving ad hoc search effectiveness. Unlike existing work, our methods exploit recommender techniques to extract the correlation between documents and apply detected correlations in a cluster-based PageRank algorithm to compute the importance of each document in a dataset. In this study two popular recommender techniques are examined in four proposed PageRank models to investigate the effectiveness of our approach. Comparison of our methods with strong baselines demonstrates the solid performance of our approach. Experimental results are reported on an extended version of the FIRE 2011 personal information retrieval (PIR) data collection which includes topically related queries with click-through data and relevance assessment data collected from the query creators. The search logs of the query creators are categorized based on their different topical interests. The experimental results show the significant improvement of our approach compared to results using standard IR and cluster-based PageRank methods.

Categories

H.3.1 [**Information Storage and Retrieval**]: Content Analysis and Indexing; H.3.3 [**Information Search and Retrieval**]: Retrieval Models

General Terms

Algorithms, Experimentation

Keywords

Information retrieval, recommender techniques, cluster-based PageRank model, Markov random model, re-ranking

1. INTRODUCTION

Information Retrieval (IR) systems aim to identify relevant information to satisfy the current user's information need. Different techniques can be exploited to attempt to improve retrieval effectiveness, including personalized information retrieval (PIR), combination of recommender systems (RSs) with IR [3][19], or using inter-document link structures via algorithms such as PageRank [9]. However, these techniques have limitations. For example, PIR systems collect both explicit and implicit feedback to build a user profile with the objective of giving retrieval results which better meet individual user information needs. However, in many situations there may be no opportunity to collect suitable feedback information to assist with the current query which is a significant challenge for PIR systems. To address this challenge, in our previous work [19], we examined integrating RS output with a standard IR in a late stage fusion method. We observe that, fusing IR and RS directly can improve the final IR result. However, since IR and RS have different goals, we believe that there may be better approaches to exploiting recommender techniques to aid IR results than simple fusion of existing methods. The PageRank algorithm is a popular and widely used method to compute the page importance in commercial Web search (e.g. Google). PageRank type methods have also used successfully in other research, such as multi-document summarization [15] and ad hoc search [9].

Based on these observations and analysis, we note that using either RSs or a PageRank algorithm can improve IR results [9][19]. We propose a novel approach to improve standard IR in this study by using RSs, PageRank and IR in a combined strategy. We utilize recommender techniques to extract the correlation between documents in the dataset, and apply this detected correlation to the PageRank algorithm to compute the importance of each document. Finally this document importance list is used to re-rank the IR output with the objective of enhancing retrieval effectiveness. The proposed methods are evaluated using the FIRE 2011 dataset. In this dataset, different topical focused category models are built, each of these includes a number of queries and search logs relating to similar search interests. We investigate two different recommender algorithms (RAs) in this work to compute the affinity weight between documents, and use these affinity weight values to compute the importance of each document in every topic category. We also propose two models that take into account the relation between documents and cluster-level information, which is the strength of the relation between each document and the current query. These factor values are applied to the cluster-based PageRank algorithm, and the results used to re-rank the IR output. Very encouraging experimental results for out methods are obtained in this study.

The remainder of the paper is organized as follows: Section 2 introduces related work, Section 3 describes our proposed integrated IR model and methods, the experiments and results are then described in Section 4, finally in Section 5, we draw conclusions and discuss our future plans to extend this work.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permission from Permissions@acm.org.

ICTIR'13, September 29-October 02, 2013, Copenhagen, Denmark.
Copyright 2013 ACM 978-1-4503-2107-5/13/09&#0133; $15.00.

2. RELATED WORK

This section reviews relevant existing work on the relationship between IR and RSs, and the application of PageRank and other link analysis methods.

2.1 IR combined with RS

Most existing work which attempts to exploit the link between IR and RSs focuses on methods which seek to reformulate the RS problem as an IR one [3][4][6]. The basic concept of doing this is to move from the RS domain to the IR domain. Each user is considered to be a document and each document rating provided by this user as a term: in this way, as in an IR model, a document is a set of terms. In the RS domain a user is characterized by a set of documents to which this user has given ratings. Moreover, the current user becomes the query to the IR system, which means that in the IR system we want documents which are more similar to the query, but for the RS problem we want users who are more similar to the current user. Beyond this point, any standard IR algorithm can be used to obtain a ranked list representing the set of users more similar to the current user, ordered by decreasing similarity. Finally, the output ranked list is used to give predictions to the current user, which again goes back to the RS domain.

2.2 PageRank for Ad Hoc Search

PageRank is one of most popular algorithms for link analysis between web pages and has been successfully used to improve web retrieval. More advanced web link analysis methods have been proposed to leverage the multi-layer relationships between web pages. The Conditional Markov Random Walk Model has been successfully applied in web page retrieval tasks based on a two-layer web graph [12]. The hierarchical structure of the web graph is also exploited for link analysis in [16]. In recent years, more researchers have focused on using link analysis methods to re-rank search results in order to improve retrieval performance [9][10][17]. The links between documents are induced by computing the similarity between documents using a cosine measure [2] or language model measures [9]. In addition, link analysis methods have also been applied in social network analysis [18], multi-document summarization [15] and other tasks. Two existing link analysis models are introduced in the following sub-sections. Based on these existing models, in Section 3 we propose a novel three-layer model to improve ad hoc search effectiveness.

2.2.1 Basic One-Layer Model

The Markov Random Walk (MRW) modelis essentially a way of deciding the importance of a vertex within a graph based on global information recursively drawn from the entire graph. The basic idea is that of “vote” or “recommendation” between vertices [15]. A link between two vertices is considered as a vote cast from one vertex to the other. The score associated with a vertex is determined by the votes that are cast for it, and the score of the vertices casting these votes.

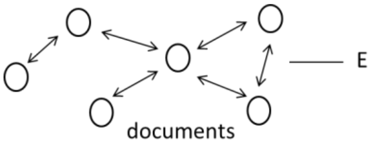

Figure 1. One-layer link graph

Formally, given a document set *S*, let *G = (V, E)* indicate a graph which reflects the relationships between documents in the document set, as shown in Figure 1. *V* is the set of vertices, each vertex v_i in *V* is a document in the document set. *E* is the set of edges, which is a subset of *V*×*V*. Each edge e_{ij} in *E* is associated with an affinity weight f *(i -> j)* between documents v_i and v_j *(i ≠ j)*. Each document v_i is represented as a set of terms $v_i(t_1,t_2,...,t_n)$. The affinity weight is computed using the standard cosine measure [2] between two documents, Equation (1)..

$$f(i \rightarrow j) = sim_{\cos ine}(v_i, v_j) = \frac{\vec{v}_i \cdot \vec{v}_j}{|\vec{v}_i| \times |\vec{v}_j|} \tag{1}$$

where $\vec{V_i}$ and $\vec{V_j}$ are the term vectors of v_i and v_j. Two vertices are connected if their affinity weight is larger than 0. Let $f(i \rightarrow i) = 0$ to avoid self-transition. The transition probability from v_i to v_j is then defined by normalizing the corresponding affinity weight, Equation (2)

$$p(i \rightarrow j) = \begin{cases} \frac{f(i \rightarrow j)}{\sum_{k=1}^{|V|} f(i \rightarrow k)} & if \sum f \neq 0 \\ 0 & otherwise \end{cases} \tag{2}$$

Formally, *p(i->j)* is not equal to *p(j->i)*. in [15], $(M_{i,j})_{|V|\times|V|}$ is used to describe *G* with each entry corresponding to the transition probability $M_{i,j} = p(i \rightarrow j)$. In order to make *M* into a stochastic matrix, the rows with all zero elements are replaced by a smoothing vector with all elements set to $1/|V|$. However, in this study, for our search task, described in Section 3.1, the direction of the relation between two documents is not considered, which means that $p(i \rightarrow j)$ is equal to $p(j \rightarrow i)$. The same technique is used to make *M* into a stochastic matrix. The saliency score for document v_i can be deduced from matrix *M* and formulated in a recursive form as in the PageRank algorithm, shown in Equation (3).

$$Score(v_i) = \lambda \cdot \sum_{j \neq i} Score(v_j) M_{j,i} + \frac{(1-\lambda)}{|V|} \tag{3}$$

where λ is a damping factor usually set to 0.85, as in the PageRank algorithm [14]. For implementation, the initial scores of all documents are set to 1 and the iteration algorithm in Equation (3) is adopted to compute the new scores of the documents. The convergence of the iteration algorithm is achieved when the difference between the scores computed at two successive iterations for any documents falls below a given threshold, the threshold is set to 0.001 in this study.

2.2.2 Two-Layer Model

A cluster-based conditional MRW model was proposed in [15], this conditional MRW model is based on a two-layer link graph including both documents and clusters. This work posited that a document set usually contains a few topic themes, and that each theme can be represented by a cluster of topic-related sentences, but that the theme clusters are not equally important. In [15], three popular clustering algorithms were explored to detect theme clusters within the document set: K-means clustering, agglomerative clustering and divisive clustering.

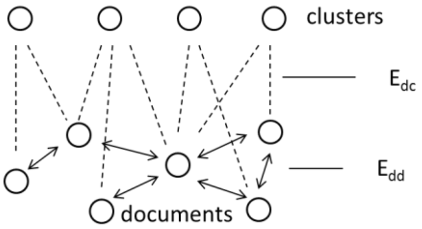

Figure 2. Two-layer link graph

The link representation is shown in Figure 2. The lower layer represents the traditional link graph between documents with the upper layer representing the theme clusters. The dashed lines between these two layers indicate the conditional influence between the documents and the clusters.

Formally, this new representation of the two-layer graph is denoted as $G^* = (V, V_c, E_{dd}, E_{dc})$, where V is the set of documents, V_c is the set of hidden nodes representing the detected theme clusters, $E_{dd}=\{e_{ij}| \ v_i, \ v_j \in V\}$ corresponds to all links between documents, and $E_{dc}=\{e_{ij}| \ v_i \in V, \ c_j \in V_c \text{ and } c_j=C(v_i)\}$ corresponds to the correlation between a document and its cluster. $C(v_i)$ indicates the theme cluster containing document v_i. [15] incorporates two factors, source cluster $C(v_i)$ and destination cluster $C(v_j)$, into the transition probability from v_i to v_j; the new transition probability is defined as follows:

$$p(i \to j \mid C(v_i), C(v_j)) = \begin{cases} \dfrac{f(i \to j \mid C(v_i), C(v_j))}{\sum_{k=1}^{|V|} f(i \to k \mid C(v_i), C(v_k))} & if \sum f \neq 0 \\ 0 & otherwise \end{cases} \tag{4}$$

where the $f\left(i \to j|C(v_i), C(v_j)\right)$ is the affinity weight between two documents v_i and v_j, conditioned on the two clusters containing the two documents. $f\left(i \to j|C(v_i), C(v_j)\right)$ is computed as shown in Equation (5).

$$\begin{aligned} & f(i \to j \mid C(v_i), C(v_j)) \\ & = \beta \cdot f(i \to j \mid C(v_i)) + (1-\beta) \cdot f(i \to j \mid C(v_j)) \\ & = \beta \cdot f(i \to j) \cdot \pi(C(v_i)) \cdot \omega(v_i, C(v_i)) \\ & + (1-\beta) \cdot f(i \to j) \cdot \pi(C(v_j)) \cdot \omega(v_j, C(v_j)) \\ & = f(i \to j) \cdot (\beta \cdot \pi(C(v_i)) \cdot \omega(v_i, C(v_i)) \\ & + (1-\beta) \cdot \pi(C(v_j)) \cdot \omega(v_j, C(v_j))) \end{aligned} \tag{5}$$

where $\beta \in [0,1]$ is the combination weight controlling the relative contributions from the source cluster and the destination cluster. Further, let $\pi\left(C(v_i)\right) \in [0,1]$ denote the importance of cluster $C(v_i)$ in the whole document set S,. This aims to evaluate the importance of the cluster $C(v_i)$ in document set S, and is computed by the cosine similarity value between the cluster and whole document set:

$$\pi(C(v_i)) = sim_{\cos ine}(C(v_i), S) \tag{6}$$

$\omega\left(v_i, C(v_i)\right) \in [0,1]$ denotes the strength of the correlation between document v_i and its cluster $C(v_i)$. This aims to evaluate the correlation between the document v_i and its cluster $C(v_i)$, and is set to the cosine similarity value between the document and the cluster:

$$\omega(v_i, C(v_i)) = sim_{\cos ine}(v_i, C(v_i)) \tag{7}$$

The new row-normalized matrix M^* is defined as:

$$M_{i,j}^{\ *} = p(i \to j \mid C(v_i), C(v_j)) \tag{8}$$

the saliency score for each document is computed based on the matrix M^* by using the iterative form in Equation (3).

3. PROPOSED MODELS

In our previous work [19], we proposed an enhanced IR model which incorporates IR with RSs. This method records all users' search behaviour and categorizes these user logs into different categories based on their different topic focus, we refer to each category as a topic category. For a new input query, we obtain a ranking from an IR system, but also select the best matching topic category for this query. The method of selecting the topic category is introduced in the Section 4.2, and outputs a set of predicted ranked results using an RA based on the selected topic category. Finally, we combine the predicted ranking result with the IR component output.

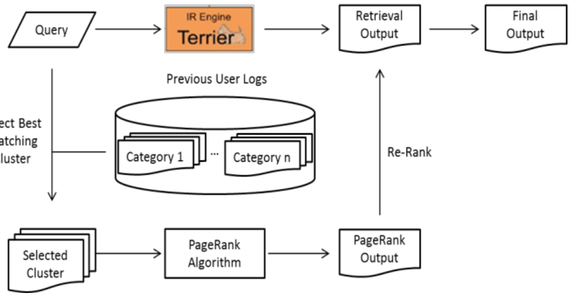

Figure 3. Basic framework of proposed retrieval model.

Figure 3 shows the basic framework of the integrated IR model proposed in this study. The difference between this study and our previous work [19], is that instead of using a recommender technique for the recommender component, we use a PageRank algorithm to calculate the document importance ranking, and combine this with the ranked IR output.

The "Previous User Logs" data set, shown in Figure 3, contains a set of topic categories. As mentioned, each topic category includes user search logs of similar interests. Every user's search log includes a list of viewed documents and the rating they gave to each document. In practice, users may be unwilling to provide ratings for viewed documents, However, [13] demonstrates that the length of time which a user stays on a document is a good indicator of the quality and its importance to them. Thus we calculate a document rating by extracting the dwell time that this user spent on each document. This results in each cluster containing a number of weighted documents. We assume that if any two documents are rated by the same user and their affinity weight is larger than 0, that there is a connection between these two documents. Formally, as introduced in Section *2.2.1*, PageRank detects the affinity weight between documents by computing a similarity between them, and exploits this affinity weight to predict each document's importance. In our method, as mentioned, we extract the dwell time to be used as the rating value for each document. Based on the observations in [13], instead of using cosine similarity, shown in Equation (1), we exploit recommender techniques which use the rating value of each document to compute the similarity between documents, to extract affinity between documents, and use this extracted correlation to compute the importance of each document.

Two recommender techniques are investigated in this study: adjusted cosine similarity [5] and weighted slope one [11]. We need to select the best matching topic category for each query, we observe that this category selection process introduces another affinity between the current query and every topic category. Based on this observation, besides using the two recommender techniques in the two-layer PageRank model, another two methods are proposed by incorporating this query-cluster level affinity into them. Thus, four methods are proposed in total, the details of these methods are introduced in the following sub-sections.

3.1 Adjusted Cosine Similarity

Based on the cluster-based conditioned two-layer model introduced in Section 2.2.2, instead of using cosine similarity to compute the affinity weight between documents, in this section, we propose a method by exploiting adjusted cosine similarity [5] to compute the affinity weight. The adjusted cosine similarity is used to compute the similarity between two documents based on the ratings they received from different users for every document. As introduced above, each topic category contains a number of weighted documents, the affinity weight between documents can simply be computed by exploiting the weight of documents in all topic categories. It is computed as the mean adjusted cosine similarity value of topic categories which contain both document v_i and v_j. For the current query q, one best match topic category C_q is then selected (the detail of the method of categorizing the current query is introduced in Section 4.2). The affinity weight between any two documents v_i and v_j in topic category C_q is compute using Equation (9).

$$f(i \to j) = \frac{\sum\limits_{C(v_i,v_j)} \frac{\sum_{u \in U}(v_{ui} - \overline{v_u})(v_{uj} - \overline{v_u})}{\sqrt{\sum_{u \in U}(v_{ui} - \overline{v_u})^2 \sum_{u \in U}(v_{uj} - \overline{v_u})^2}}}{card(C(v_i,v_j))} \tag{9}$$

where v_{ui} is the rating user u gives to document v_i, $\overline{v_u}$ indicates the average rating of user u who rates both documents v_i and v_j in the selected topic category, U is the set of users in the corresponding topic category. $C(v_i,v_j)$ is the category containing both document v_i and v_j. $card(C(v_i,v_j))$ denotes the number of topic categories which contain both document v_i and v_j. For the current query q, in the selected topic category C_q, the transition probability from v_i to v_j is then computed using Equation (10).

$$p(i \to j \mid C_q(v_i,v_j)) = \begin{cases} \dfrac{f(i \to j \mid C_q(v_i,v_j))}{\sum\limits_{k=1}^{|V|} f(i \to k \mid C_q(v_i,v_k))} & if \sum f \neq 0 \\ 0 & otherwise \end{cases} \tag{10}$$

where $f(i \to j|C_q(v_i,v_j))$ is caclcuated using Equation (11).

$$\begin{aligned} & f(i \to j \mid C_q(v_i,v_j)) \\ & = \beta \cdot f(i \to j) \cdot \omega(v_i,C_q) + (1-\beta) \cdot f(i \to j) \cdot \omega(v_j,C_q) \\ & = f(i \to j) \cdot (\beta \cdot \omega(v_i,C_q) + (1-\beta) \cdot \omega(v_j,C_q)) \end{aligned} \tag{11}$$

where $\beta \in [0,1]$ is the combination weight. Let $\omega(v_i,C_q) \in [0,1]$ indicate the strength of the correlation between document v_i and the topic category C_q, which is computed by $\omega(v_i,C_q) = sim_{cosine}(v_i,C_q)$. The new row-normalized matrix M^* is defined as shown in Equation (12).

$$M_{i,j}^{*} = p(i \to j \mid C_q(v_i,v_j)) \tag{12}$$

the saliency score for each document is computed based on the matrix M^* by using the iterative form shown in Equation (3).

3.2 Conditional Adjusted Cosine Similarity

This section introduces our method to incorporate the query-level information and query–to-cluster relationship into the adjusted cosine similarity method described in Section 3.1. We observe that the query–to-cluster relationship is an indicator of the relevance between documents and the current query, based on this observation, our novel approach is shown in Figure 4.

The lower layer shown in Figure 4 is the traditional link graph between documents in the basic MRW model, The link between documents contains two kinds of correlations: $E_{dd}=\{e_{ij}| v_i, v_j \in V\}$ corresponds to the links between documents, and $N_{dd}=\{e_{ij}| card(u_{i,j})\}$ is the number of users who rate both documents v_i and v_j. The middle layer in the figure represents the topic categories, the dashed lines between lower and middle layers indicate the conditional influence between the documents and the topic categories. The upper layer is the query layer, the dashed lines between the query layer and the topic category layers indicate the strength of the correlations between queries and topic categories.

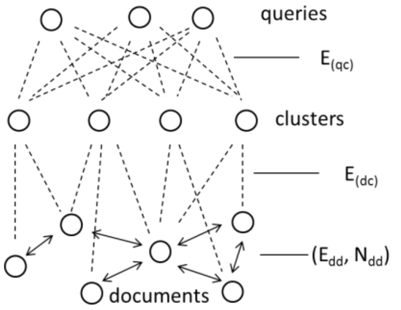

Figure 4. Three-layer link graph

We suggest a hypothesis that there is a strong correlation between any two documents rated by the same user. Based on this hypothesis, we adopt the factor N_{dd} to indicate the strength of the correlation between two documents. $E_{(qC)}$ denotes the correlation between a query q_i and each topic category C. Then for the current query q, withint its selected best match topic category C_q, the affinity weight between any two documents v_i and v_j is computed as shown in Equation (13).

$$f(i \to j) = \frac{\sum\limits_{C(v_i,v_j)} \frac{\sum_{u \in U}(v_{ui} - \overline{v_u})(v_{uj} - \overline{v_u})}{\sqrt{\sum_{u \in U}(v_{ui} - \overline{v_u})^2 \sum_{u \in U}(v_{uj} - \overline{v_u})^2}}}{card(C(v_i,v_j))} \times N(v_i,v_j) \tag{13}$$

where $N(v_i,v_j)$ denotes the number of users who rate both documents v_i and v_j in all topic categories. So for the current query q, in the selected topic category C_q, the transition probability from document v_i to document v_j is computed using Equation (14).

$$p(i \to j \mid q,C_q(v_i,v_j)) = \begin{cases} \dfrac{f(i \to j \mid q,C_q(v_i,v_j))}{\sum\limits_{k=1}^{|V|} f(i \to k \mid q,C_q(v_i,v_k))} & if \sum f \neq 0 \\ 0 & otherwise \end{cases} \tag{14}$$

where $f(i \to j|q,C_q(v_i,v_j))$ is calculated using Equation (15).

$$\begin{aligned} & f(i \to j \mid q,C_q(v_i,v_j)) \\ & = \beta \cdot f(i \to j) \cdot \pi(q,C_q) \cdot \omega(v_i,C_q) \\ & + (1-\beta) \cdot f(i \to j) \cdot \pi(q,C_q)) \cdot \omega(v_j,C_q) \\ & = f(i \to j) \cdot \pi(q,C_q) \cdot (\beta \cdot \omega(v_i,C_q) + (1-\beta) \cdot \omega(v_j,C_q)) \end{aligned} \tag{15}$$

where $\beta \in [0,1]$ is the combination weight. $\pi(q,C_q) \in [0,1]$ denotes the strength of correlation between the current query q and the selected cluster C_q, and is computed by: $\pi(q,C_q) = sim_{cosine}(q,C_q)$, also set $\omega(v_i,C_q) \in [0,1]$. The new row-normalized matrix M^* is defined as shown in Equartion (16).

$$M_{i,j}^{*} = p(i \to j \mid q,C_q(v_i,v_j)) \tag{16}$$

The saliency score for each document is computed based on the matrix M^* by using the iterative form in Equation (3).

Table 1. Examples of user behaviour data

User ID	Topic	query	docID	Time
Test1	Indian armed forces	weaponry	en.15.204.215.2007.8.22	2011-08-23 10:33:06
Test1	Indian armed forces	weaponry	1040715_foreign_story_3498066.utf8	2011-08-23 10:34:20
Test1	Indian armed forces	weaponry	1040715_foreign_story_3498066.utf8	2011-08-23 10:36:02
… …	… …	… …	… …	… …

3.3 Document Deviation

Similar to the of adjusted cosine similarity to compute the correlation between documents in Section 3.1, in this section, we exploit another recommender technique. This is a rating-based RA which we use to extract the correlations between documents based on the ratings assigned to them. We use the simple, popular and effective rating-based weighted Slope One Scheme [11] algorithm to extract the correlation between documents. Affinity weight $f(i \to j)$ is defined as the deviation between documents v_i and v_j instead of the similarity between two documents. This deviation is a measurement of the difference between the documents. This difference is based on the mean average of the users' ratings of the documents. Both document deviation and the number of users' search logs which contain a rating for both documents v_i and v_j are used to represent the correlation between documents v_i and v_j, and are explored in this method to improve effectiveness of the MRW model. The previous user logs dataset is used as a training set χ, with any two documents v_i and v_j with ratings u_i and u_j respectively in some user's evaluation u (annotated as $u \in S_{i,j}(\chi)$), to compute the average deviation of document v_i with respect to v_j using Equation (17).

$$f(i \to j) = dev_{v_i,v_j} = \sum_{u \in S_{i,j}(\chi)} \frac{u_j - u_i}{card(S_{i,j}(\chi))} \qquad (17)$$

where $card(S_{i,j}(\chi))$ indicates the number of users who rate both document v_i and v_j in all topic categories. The new transition probability $p(i \to j|C_q(v_i, v_j))$ is computed using Equation (10). $f(i \to j|C_q(v_i, v_j))$ is computed using Equation (11). The new row-normalized matrix M^* and the saliency score for each document are defined and computed in the same way as in Section 3.1.

3.4 Conditional Document Deviation

This section introduces a method which combines Section 3.2 with Section 3.3 by exploiting the document deviation to compute the affinity weight between documents in the three-layer model introduced in Section 3.2. The affinity weight $f(i \to j)$ between document v_i and v_j is computed using Equation (17). The new transition probability $p(i \to j|q, C_q(v_i, v_j))$ is computed using Equation (14). The new row-normalized matrix M^* and the saliency score for each document are defined and computed the same way as described in Section 3.2.

4. EXPERIMENTS AND RESULTS

4.1 Data Set

The user behaviour data collected for the FIRE 2011 PIR task [8] is used to evaluate the techniques introduced in Section 3.. 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, comprising almost 400k documents in total.

This dataset contains the user search log information collected from a number of volunteer users. It is an ideal dataset to explore our proposed method to utilize previous users search information to compute the relevance of each document to improve the IR results. The following sub-sections overview the steps of creating the PIR test collection.

User Behaviour Data

The following steps were carried out to collect users search behaviour information:

- Participants volunteered to search the document collection in one of 27 provided news topic areas. Each participant selected one of the 27 topics themself to ensure that this was an area in which they were knowledgeable and interested. They then created a topic statement (query) related to the chosen topic.
- The participant then submitted their query to an IR system which returned a ranked list of potentially relevant news documents. The participant then began viewing document snippets from the list and could click a document to reveal its full contents. They continued this until they found the information they needed or gave up the search. The participant's activities were tracked and logged. The log recorded information including participant's username, the topic category selected, the contributed query, the returned documents viewed, and the dwell time spent on each document. Table 1 shows examples of topic categories and the structure of the user behaviour data in each topic category. Although the dwell time of each document can depend on the document length, since the documents in this collection are relatively short, we regard the dwell time as a reasonable measure of expected document relevance.
- In addition, each participant was also required to provide relevance assessments for the queries they provided. They were asked to read the top 30 documents in the ranked list returned for each of the query they entered, and to mark relevant documents which addressed their queries. These selected relevant documents were used as the relevance assessment data for experiments. Note that this relevance assessment collection session was separate from the search log collection procedure.

In total, 26 participants contributed 150 queries for the 27 topics. It should be noted that since the participants were given free choice of topic, that the queries are distributed unevenly over the available topics. One query was randomly selected from each topic category to be used as test query for this topic. This resulted in 123 queries to be used as a training set for the RS and

27 queries to be used as the test topic set. All parameters in this study were trained empirically using this training set.

From the participants search information shown in Table 1, “user ID” is user’s ID information, “Topic” indicates the topic area, “query” is the query user insert, “docID” denotes the documents this user viewed and “Time” is the dwell time information for every document from this user. The user log information was processed to extract staying time information and build a link graph between documents.

Staying time extraction

In this work, we segmented the viewing session based on the query. If the query changed, a new session was started. For every viewing session, we used the difference between the time of the second document and that of the first document as the observed staying viewing time on the first document. For example in Table 2, the part of searcher test1’s search log, we set the time 10:34:20 as 1034.20, the view time for document ‘en.15.204.215.2007.8.22’ was calculated as *1034.20 - 1033.06 = 1.14*. For the last document in a session, we used the following heuristics to decide its observed staying time, we computed the average viewing time from the distribution of observed viewing time of documents in all the records of this user and took this as the observed viewing time for the document.

After extracting the staying time information, we built a user search log for the query, and clustered the user logs into different topic categories based on the “Topic” they chose. For each topic category, we collected the different user’s search logs. Figure 5 shows the previous user logs data and the structure of user logs in *N* different topic categories.

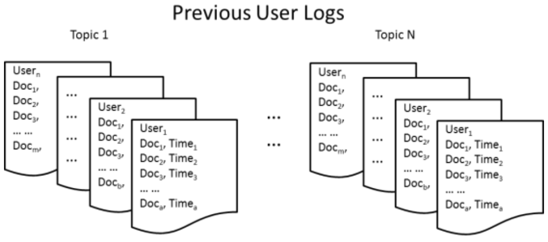

Figure 5. Previous user search logs data include different topic categories and the structure of user logs in every topic category

Link graph construction

The processed user log information was used to build a link graph for each topic category. Our assumption is that in every topic category, if any two documents have been rated by the same user and the affinity weight between them is greater than 0, that there is a correlation between them, and a link is built between them. This correlation does not have direction. Each topic group can be seen as a topical focused category. Every topic category can generate a user browsing graph like that shown in Figure 6. The different types of lines in Figure 6 represent the link graph for different users.

4.2 Experiment Setup

4.2.1 IR Component

The Terrier BM25 retrieval model [20], shown in Equation (18), was used to generate ranked lists for the IR component. A stopword list containing of 500 words was used with a Porter stemmer to preprocess the input text. A standard TREC formatted ranked list of 1000 documents was returned for each query.

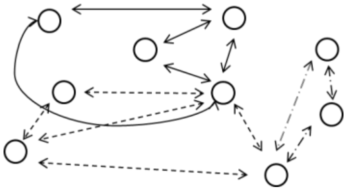

Figure 6. Sample of user browsing graph in one topic category

$$score(D,Q)=\sum_{i=1}^{n} IDF(q_i)\cdot\frac{f(q_i,D)\cdot(k_1+1)}{f(q_i,D)+k_1\cdot(1-b+b\cdot\frac{|D|}{avgdl})} \qquad (18)$$

where query *Q* contains a set of keywords $\{q_1,q_2,...,q_n\}$, *score(D,Q)* is the relevance score between query *Q* and document *D*, $IDF(q_i)$ is the inverse document frequency weight of query term q_i. $f(q_i,D)$ denotes the frequency of query term q_i in document *D*, and *avgdl* is the average length of documents in the collection . k_1 and *b* are free parameters, usually k∈[1.2, 2.0] and *b=0.75*.

4.2.2 Centroid Document Generation

When computing the similarity between the query and topic category, problems arise since the length of both the query and topic description are usually too short to compute the similarity reliably. To address this problem, we generate a centroid documents to represent the query and each topic category.

In order to generate the centroid document for the current query, we used the retrieval results obtained from Terrier for the current query. Similar to blind relevance feedback (BRF), we assume that the top *N* documents in the retrieved ranked list to be relevant and use them to generate the centroid representation for the query. First, we take top *N* documents on the retrieval list into a set *S*, then for each document *d* in *S,* stopwords are first removed with subsequent application of Porter stemming, The resulting document vector is then weighted using TF-IDF to produce a weighted vector $d_{tf\text{-}idf}$= ($tf\text{-}idf_1$, $tf\text{-}idf_2$,…), where $tf\text{-}idf_i$ is the term frequency inverse document frequency of the *i*th term in document *d*. For the document set *S,* we define its centroid document *C* in Equation (19).

$$Centroid(S)=\frac{1}{|S|}\sum_{d\in S}d \qquad (19)$$

where *|S|=N=5* in this study, which was set empirically based on the training set. and $Centroid_q$ refers to the centroid document for the query *q*. A similar method is used to generate the centroid document for each topic category: The document frequency for each document in every topic category was computer, and all documents were ranked in descending order of their frequency in the topic category. The top 5 documents which occur most times in each topic category, were used to generate the centroid document [7] for the corresponding topic category using Equation (19). The generated centroid document is assumed to the represent the corresponding topic category. Here the number of top 5 was again chosen empirically; top 3 and top 10 were also examined, with the top 5 performing best.

4.2.3 Categorizing

The purpose of this step is to attempt to identify the correct topic category for each test query. From the dataset, we observe that the queries which were created by real world users are usually very short. The very short length of these queries means that topic category selection can be unreliable. Our earlier experiments showed that the accuracy of topic category selection can be improved by expanding the query and topic category descriptions, and we adopt this approach in this investigation. So the categorizing process is:

- Generate the centroid documents for both current query and each topic category to expand both short length query and short length topic for each topic category.
- Match the query representation (query centroid document) to each topic category by using cosine function (Equation (20)) to compute the distance between them. The closest topic category was selected for the current query q.

$$Similarity(C_q, C_{topic}) = \frac{C_q \cdot C_{topic}}{\|C_q\|\|C_{topic}\|} \tag{20}$$

As introduced in Section 4.2.2, when generating the centroid document for each topic category, the top 5 most occurred documents in that topic category are chosen. The reason that the correct topic category is not selected on some occasions is that we simply use the 5 highest frequency documents to generate the 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, only using the highest frequency documents to build the centroid may lead to topic drift. We will examine methods which seek to improve this method in our future work.

4.2.4 Re-ranking

The output of our PageRank model was used to re-rank the retrieval results for each test query. We use the combSUM operator to combine results for re-ranking. Let S_{IR} refer to the retrieval list and S_{PR} to our novel PageRank output list. For the combSUM method, every document's new weight was calculated as follows using Equation (21).

$$d'_{SUM} = \begin{cases} d_{IR} + d_{PR} & if\ (d \in S_{IR} \cap S_{PR}) \\ d_{IR} & otherwise \end{cases} \tag{21}$$

where d_{IR} refers the relevance score of document d in S_{IR}, d_{PR} indicates the importance score of document d in the S_{PR} ranking.

4.3 Results

In this experiment, Mean Average Precision (MAP) was used to evaluate overall retrieval effectiveness and precision at cut-off n to evaluate how early the relevant documents were retrieved. Four different baselines were used to compare with our approaches:

- Initial standard retrieval results (IR).
- Query expansion using the Terrier default Bo1 pseudo-relevance feedback (PRF) for query expansion, which is a term weighting model based on Bose-Einstein statistics and is similar to Rocchio. In Bo1, the informativeness $\omega(t)$ of a term t is given by Equation (22). The query expansion process takes the top 5 documents in the initial retrieval list to extract expansion terms by computing the term weight $\omega(t)$, 5 terms to the test query. Use of the top 5 documents and addition of 5 terms was selected empirically (IR+QE).

Table 2. Comparison of retrieval results for all methods

	MAP	P@5	P@10	P@20
IR	0.1225	0.1173	0.0947	0.0653
IR+QE	0.1470 (+20.0%)	0.1360 (+15.9%)	0.0880 (-7.07%)	0.0673 (+3.06%)
IR+WS1	0.1811 (+47.8%)	0.1569 (+33.8%)	0.1296 (+36.9%)	0.0857 (+31.2%)
IR+PR	0.2207* (+80.2%)	0.1947* (+65.9%)	0.1173* (+23.9%)	0.0913* (+39.8%)
Acos_PR	0.2310*† (+87.8%)	0.2000* (+70.5%)	0.1373*† (+44.9%)	0.0900* (+37.8%)
CAcos_PR	0.2454*† (+100%)	0.2027*† (+72.8%)	**0.1387*†** **(+46.0%)**	0.0851* (+30.5%)
Dev_PR	0.2498*† (+103%)	0.2005* (+70.9%)	0.1360*† (+41.5%)	**0.1040*†** **(+40.9%)**
CDev_PR	**0.2593*†** **(+112%)**	**0.2035*†** **(+73.5%)**	0.1384*† (+45.7%)	**0.1040*†** **(+40.9%)**

$$\omega(t) = tf_t \cdot log_2 \frac{1 + P_n}{P_n} + log_2(1 + P_n) \tag{22}$$

where tf_t is the frequency of the term t in the pseudo-relevant set (top N document set) and P_n is given by $F/|S|$. F is the term frequency of the query term in the whole collection and $|S|$ is the number of the documents in the collection.

- Adopt rating-based Weighted Slope One Recommender algorithm to compute prediction list, combine this prediction results with IR results by combSUM (IR+WS1).
- PageRank algorithm (IR+PR) results: in this method each document is represented as a bag of terms, using cosine similarity (Equation (1)) to compute the affinity weight between documents in the selected topic category. Then the PageRank algorithm (Equation (3)) is used to output the document importance ranking. Finally, the IR output is re-ranked using the PageRank results with the combSUM operator.

Our four methods were introduced in Section 3. In summary they are:

- Using adjusted cosine similarity in a two-layer cluster-based PageRank algorithm (Acos_PR) to re-rank IR results
- Using adjusted cosine similarity in the conditioned three-layer cluster-based PageRank algorithm (CAcos_PR) to re-rank IR results
- Using a rating-based recommender technique to compute the deviation between documents, and applying this deviation correlation in two-layer cluster-based PageRank model (Dev_PR) to re-rank IR results
- Finally, the deviation correlation in the conditioned three-layer cluster-based PageRank method (CDev_PR) is used to re-rank IR results.

From Table 3, we can observe that the CDev_PR performs best among these methods. IR+PR, Acos_PR, CAcos_PR, Dev_PR and CDev_PR methods achieved statistically significant improvements over standard IR results (marked with *). Acos_PR, CAcos_PR, Dev_PR and CDev_PR offer statistically significant improvements over IR+PR marked with †. We observe from Table 3 that the CDev_PR method outperforms other methods, achieving significant improvements in MAP, P@5, P@10 and P@20 over both the results of IR and IR+PR.

From these results, we can see that the correlation between the current query and clusters, the clusters and documents are useful factors in improving the results of PageRank methods. Further, combining the obtained improved PageRank output with standard IR results can augment the retrieval results. Also, utilizing recommender techniques to compute the affinity weight between documents in the cluster-based PageRank algorithm to re-rank IR output can improve the results of standard IR algorithms

5. CONCLUSION AND FUTURE WORK

In this paper, we proposed to adopt recommender techniques to detect correlations between documents in a cluster-based PageRank model. We also introduced a three-layer cluster-based PageRank model. We track and record users' search behaviour, and categorize the users' search information into different topical category. A link graph is built for each topic category, and recommender techniques are exploited to compute the affinity weight between documents, these correlations are applied to PageRank algorithm, and finally an output list of the importance of documents in the selected topic category is generated. The document importance scores are utilized to re-rank the standard IR output. The proposed methods have been compared with standard blind relevance feedback query expansion, a standard PageRank algorithm and recommender techniques applied directly to the IR system. Results show that our methods perform more effectively than these runs. We conclude that exploiting recommender features in the PageRank algorithm can improve over standard IR system results.

We examined two recommender techniques in this study, both of them based on user ratings. In the future, we plan to examine the performance of other recommender methods, such as content-based recommender algorithms or item-based approaches. In our earlier work [19], we combined recommenders algorithm with a standard IR system, this work shows that applying recommender techniques into the existing mature PageRank methods obtains better results. In this work, we only examine our method on a relatively small collection, in future we hope to evaluate our methods on much larger data collections to further examine their effectiveness. In this work, the topic categories are predefined and finely described, in future work, we plan to investigate the effectiveness of the integrated model when the predefined topic categories have poor coverage of users' queries or the definition of the topics is more coarse, in order to examine the generality of our conclusions.

6. 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.

7. REFERENCES

[1] Agichtein E., Brill E. and Dumais S., Improving web search ranking by incorporating user behaviour information. In *Proceeding of SIGIR 2006: 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*. Seattle, Washington (2006)

[2] Baeza-Yates R. and Ribeiro-Neto B., Modern Information Retrieval. *ACM Press*. (1999)

[3] Belkin N.J. and Croft W.B., Information Filtering and Information Retrieval: Two sides of the same coin? *In the Magazine of Communiaction of the ACM- Special Issue on Information Filtering, Volume 35, Issue 12*. (1992)

[4] Bellogin A. and Wang J., Text Retrieval Methods for Items Ranking in Collaborative Filtering. In *Proceedings of ECIR 2011: The 33rd European Conference on Information Retrieval*. Dublin, Ireland. (2011)

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

[6] Costa. A. and Roda F.: Recommender Systems by means of Information Retrieval. *In Proceeding of WIMS 2011, International Conference on Web Intelligence, Mining and Semantics*. Sogndal, Norway. (2011)

[7] Eui-Hone H. and George K.: Centroid-Based Item Classification: Analysis & Experimental Results. *In the Proceeding of (PKDD'00) 4TH European Conference on Principles of Data Mining and Knowledge Discovery*. (2000)

[8] 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 Forum for Information Retrieval Evaluation (FIRE) 2011 Workshop*, 2nd-4th Dec 2011, Bombay, India (2011)

[9] Kurland O and Lee L.: PageRank without Hyperlinks: Structural Re-ranking Using Links Induced by Language Models. *In the Proceeding of Twenty-Eighth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (*SIGIR'05). Brazil. (2005)

[10] Kurland O and Lee L.: Respet my Authority! HITS without Hyperlinks, Utilizing Cluster-Based Language Models. *In the Proceeding of Twenty-Ninth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (*SIGIR'06). (2006)

[11] Lemire D. and Maclachlan A.: Slope One Predictors for Online Rating-Based Collaborative Filtering. *In Proceedings of SIAM 2005: International Conference on Data Mining*. Newport, CA. (2005).

[12] Liu T.Y. and Ma W.Y.: Webpage Importance Analysis Using Conditional Markov Random Walk. *In Proceedings of IEEE WI 2005*. (2005)

[13] Liu Y., Gao B., Liu T.Y., Zhang Y., Ma z.m., He S.Y. and Li H.: BrowseRank: Letting Web Users Vote for Page Importance. *In the Proceeding of Thirty-First Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (*SIGIR'08). Singapore. (2008)

[14] Page L., Brin S., Motwani R. and Winograd T.: The PageRank Citation Ranking: Bringing Order to The Web. *Technical Report*. Stanford Digital Libraries. (1998)

[15] Wan X.J. and Yang J.W.: Multi-Document Summarization Using Cluster-Based Link Analysis. *In the Proceeding of Thirty-First Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (*SIGIR'08). Singapore. (2008)

[16] Xue G.R., Li H., Yang Q., Zeng H.J., Yu Y. and Chen Z.: Exploiting the Hierarchical Structure for Link Analysis. *In Proceedings of Twenty-Eighth Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (*SIGIR'05). Brazil. (2005)

[17] Zhang B., Li H., Liu Y., Ji L., Xi W., Fan W., Chen Z. and Ma W.Y.: Improving Web Search Results Using Affinity Graph. *In Proceedings of SIGIR'05*. Brazil. (2005)

[18] Zhou D., Orshanskiy A. Zha H. and Giles C.L.: Co-ranking Authors and Documents in a Heterogeneous Network. *In Proceeding of IEEE ICDM 2007*. (2007)

[19] Li W., Ganguly D. and Jones G.J.F.: Enhanced Information Retrieval Using Domain-Specific Recommender Model. *In Proceeding of ICTIR'11*. (2011)

[20] http://en.wikipedia.org/wiki/Okapi_BM25