ISSN No. 0976-5697

Volume 5, No. 4, April 2014 (Special Issue)

International Journal of Advanced Research in Computer Science

RESEARCH PAPER

Available Online at www.ijarcs.info

A Modified Community Based Collaborative Approach for Web Search Personalization

Pradnya Prakash Bhagat*, Maruska Mascarenhas
Computer Engineering Department
Goa College of Engineering, Farmagudi, Goa-India

Abstract: Web search personalization is a strategy which accommodates differences between individuals and has become a vital element of modern search engines. Conventional personalization methods can prove effective only up to a certain level because of the limitation involved of treating web search as an isolated activity. Collaborative web search is an approach which tries to overcome this limitation by treating web search as a collaborative task in a community of like minded searchers. The community members can share their search experiences for the benefit of others while still maintaining their anonymity. The modified approach presented in this paper achieves community based personalization at the same time adding the benefits of reliability, efficiency and security to the web search.

Keywords: personalization; community; collaborative filtering; collaborative web search; stemming; stopwords; lexical database.

I. INTRODUCTION

In recent years, the World Wide Web has become the largest and most popular way of information sharing and communication. But considering its tremendous size, it might be a challenging task for the users to get the right information which suits their interests and needs at the right time. Even the most advanced search engines have been successful in indexing a very small portion of the web. A more imperative problem is the limited degree to which the pages which are covered can be accurately ranked with respect to a given query. A large part of this problem is due to the searcher as to the search engines. Most of the queries issued by users tend to be short, vague and ambiguous. For example a simple one word query like 'Java' can mean both of these things; the 'Java Programming Language' or the 'Java Islands in Indonesia'. Given only this query as the input to a search engine it may be possible to identify the primary information target but it is difficult to identify the exact context of the searcher. If any computer programmer types in the above query he would probably be interested in pages related to Java programming language, but for an explorer or a person related to geography will be least interested in it. His real interest might lie in pages related to Java Islands in Indonesia. Millions of documents will be returned when this query is submitted to a search engine like Google. Seeing the huge size of the number of results returned, we cannot expect the user to go on clicking on each end every link till they get the desired result. Hence the results returned to both of these users should be different although the query submitted is precisely the same. But current search engines do not work up to the mark with respect to dealing with such queries.

The solution to this lies in personalization, which is the strategy to accommodate differences between the individuals. Conventional search engines implemented personalization based on content analysis or analyzing the hyperlink structure of the web. But both of these methods have their own set of limitations and challenges. A major shortcoming of both of these approaches is that, the web search is treated as a solitary interaction between the user and search engine. They fail to identify the collaboration that exists between searchers. If properly harnessed, this potential can be of great help and enable us to employ an alternative approach to personalization overcoming the difficulties of conventional approaches.

This paper focuses on a group based personalization approach called modified collaborative web search inspired from collaborative filtering which allows the users to share their informative results to help other members of the community. The members of a community can collaborate for the benefit of others. The main areas focussed are; the efficiency of the data structure used for storing the data, the reliability of the results and the security of the system from malicious users.

II. MOTIVATION

The current era can be called as the age of discovery economy where the access to the right information at the right time can mark the difference between success and failure. A study shows that workers in any organization spend almost 30% of their time searching for information and despite their efforts fail to find the desired information [1]. A significant increase in productivity can be achieved if we make the right information accessible to the users at the right time. The current search engines are not able to deal with this challenge beyond the constraints. Hence, a meta-search engine can be developed which can filter the results returned by the underlying search engine so that it better suits the needs of the users.

We human are social beings. So, most of the activities that we carry out are always with the help of others. In other words we can say that we work in communities and collaborate to achieve our tasks more efficiently. There are many scenarios where even web search takes the form of a community oriented activity. For example students seeking for information on a weekly assignment may act as a community of members having common information needs. Similarly, employees of a company working on a common project will have similar information needs during the project span. Searches originating from the search box of a themed website will also serve as a community of people having

CONFERENCE PAPER
Two day National Conference on Innovation and Advancement in Computing
Organized by: Department of IT, GITAM UNIVERSITY Hyderabad (A.P.) India
Schedule: 28-29 March 2014

common interests. Also people with similar purchase history on an e-commerce web site may contribute to a community of people having similar purchase behavior. But current search engines fail to identify this potential. Identifying and harnessing this potential inherent in search communities can help to refine the search results returned to the users to a great extent.

III. LITARATURE SURVEY

A personalized web search [23] can provide different users with different search results or organize results differently for each user, based upon their interests, preferences, and information needs even if they submit exactly the same query. Several approaches have been tried and tested for implementing personalization in the past.

A. Approaches to Personalization:

a. Personatizaed Web Search based on Content Analysis: Conventionally, web search is being viewed as a solitary service operating in isolation to respond to the queries of individual searchers. Whenever a user types in a keyword in search engine, that query is searched into an inverted document index to search for the documents matching that query. The documents containing the most occurrences of the given keyword form the result set.

Search engines which support personalization, in addition to this, build profiles of each individual user [2]. User profiles store information about user's interests, preferences, likes, dislikes etc. It is built by utilizing information that is specific to a user and which is learnt explicitly or implicitly from users' browsing histories. Personalization is achieved by checking content similarity between returned web pages and user profiles. Explicit context [12] asks users to manually provide information about themselves or about the submitted query. Inquirus 2 [8] is an example of such type of search engine which asks users to select from a set of categories such as research paper, homepage etc. and uses the selected context categories to choose target search engines for the user's query. In contrast to this, implicit context [12] works by implicitly building the profiles from the search histories of users. The documents matching the keywords of user are re-ranked based on how well the document's categories match the user's interest profile. Examples of systems working on implicit context are Watson [10] and IntelliZap [14].

Limitations of this approach include; the additional cost of computing the user profiles. Also this approach can be effective only up to the level information about documents it has access to.

b. Personalized Web Search based on Hyperlink Structure of the Web: This approach works by exploring the hyperlink structure of the web. The motivation behind this approach is the recursive notion that important pages are those linked to by many other important pages. A widely popular algorithm following this approach is the PageRank algorithm implemented in initial days by Google [13] [18].

But this approach can turn out to be computation expensive since it requires multiple scans of the web graph, which makes it impossible to carry out online in response to a user query. Also when a large number of users employ a search engine, it is difficult to compute and store so many personalized PageRank vectors offline.

c. Personalized Web Search based on User Groups: In both of the above approaches, web search is considered as a solitary activity. It takes the form of an isolated interaction between the individual searcher and the search engine. However studies have shown that in most of the scenarios, information search has distinctly a collaborative flavour. There is a great amount of overlap between the information need of searchers. One approach that attempts to harness this potential existing in groups is known as collaborative filtering and is gaining increasing interest in recommendation systems. Collaborative Filtering is explained in brief in the next section.

B. Collaborative Filtering:

Collaborative Filtering is defined as the process of filtering or evaluating items based on the opinions of other people. The fundamental assumption it holds is that if two people rate on n similar items similarly then and hence will rate or act on other future items similarly.

Table: 1 a record of user versus items

	Twilight	**Harry Potter**	**Emma**	**Pride and Prejudice**
User 1	Like	Like	Dislike	Dislike
User 2	Like	Like	Like	Dislike
User 3	Like	Dislike	Like	Like
User 4	Like	?	Dislike	Dislike

Table I is a part of the database showing records of which users have liked which novels. As can be seen the prediction has to be made for User 4 to recommend him a list of possible novels that he might like. Now, to decide whether User 4 will like Harry Potter or not we search for other users who have similar liking history like User 4. For example in this case it's User 1. If sufficient records are available we can predict that even User 4 will like Harry Potter. This stands the basic principle of Collaborative Filtering. Collaborative Filtering can be classified into two types based on the similarity computation followed [22]:

a. Item based Collaborative Filtering: It performs prediction by calculating the similarity between two items [4]. To compute similarity, first it finds users who have rated both of these items and then applies similarity calculation between the two co-rated items.

b. User based Collaborative Filtering: It utilizes the similarity computed between the active user and all the other users. It considers that users who gave close rating to the same set of items have higher similarity whereas users who have different ratings for same items are less similar.

Approaches to implement collaborative filtering can be broadly classified into the following types:

c. Memory based Collaborative Filtering: These algorithms use entire or a sample of the user-item database to generate a prediction. A cluster of nearest neighbors is found for each user based on the similar interests [22]. Based on the neighbors of a new user (or active user), a prediction of recommending new items for him or her can be generated.

d. Model based Collaborative Filtering: A model can be built using machine learning, data mining

CONFERENCE PAPER
Two day National Conference on Innovation and Advancement in Computing
Organized by: Department of IT, GITAM UNIVERSITY Hyderabad (A.P.) India
Schedule: 28-29 March 2014

algorithms, which the system can use to recognize complex patterns based on training data, and then make intelligent predictions for the collaborative filtering tasks for test data or real-world data, based on learned models [9]. Some of the commonly used model based algorithms to state are; Bayesian Models, Clustering Models, Dependency networks etc.

e. ***Hybrid Collaborative Filtering:*** Hybrid Collaborative Filtering [20] combines collaborative filtering with some content based techniques to improve the quality of predictions or recommendations. Both, Collaborative Filtering and Content Analysis have some limitations and therefore cannot provide very high performance independently. Hence Hybrid Collaborative Filtering technique tries to address these problems making use of both.

Although Collaborative Filtering helps us achieve personalization by overcoming the limitations of the content based and hyperlink structure based approaches, still it has some serious disadvantages of itself. Among many other limitations, the major ones are listed below:

f. ***One-to-One Similarity Calculations:*** Similarity Metrics like Pearson's correlation coefficient are used to calculate similarity between two items or users. But this similarity calculation is one to one. I.e; it is calculated only between a pair of users or items at a time. For small data sets this method works fine, but as the size of the data set increases, the time complexity also increases to a great extent.

g. ***Privacy Violation:*** In normal collaborative filtering, a record is kept of which user selected which items; or how much rating a user gave to each of the items. Users might consider this method as violating their privacy.

Because of these drawbacks there are serious limitations when it comes to the use of collaborative filtering in web search personalization where the user base is very large and also the users prefer to stay anonymous. To deal with these problems, a modified collaborative filtering approach is proposed in [5] [6] called community based collaborative web search.

Collaborative Web search is based on the principle of collaborative filtering, but instead of exploiting the graded mapping between users and items, it exploits a similar relationship between queries and result pages. It can work as a meta-search working on an underlying search engine and re-rank the results returned by the underlying search engine based on the learned preferences of the community of users. The approaches adopted in literature for collaborative information retrieval can be distinguished in terms of two dimensions: Time and place.

With respect to time, the search can be either synchronous or asynchronous. Synchronous search sessions require the users to establish a well defined search sessions where all the users have to participate whereas asynchronous approach gives freedom to its users to search as per their convenience and still collaborate. With respect to place, the search can be either co-located or remote. Co-located approach requires the users to operate from a single PC or a single location whereas remote search can allow the users to search from two different corners of the world and still collaborate. CoSearch [21] is an example of co-located, synchronous approach. SearchTogether [16] is an example of system supporting remote search collaboration (whether synchronous or asynchronous).

A search engine built on this idea is I-Spy [11] which is based on two principle ideas: First, specialized search engines attract communities of like minded searchers with similar information needs and so serve as a useful way to limit variations in search context; and second, by monitoring user selections for a query it is possible to build a model of query-page relevance based on the probability that a given page will be selected by a user when returned as a result for query. It personalizes the search results for a community of users but does not rely at all on context-analysis. [19] explains an alternative approach to collaborative web search based on peer-to-peer network.

IV. THE MODIFIED COLLABORATIVE APPROACH FOR WEB SEARCH PERSONALIZATION

The modified collaborative approach harnesses the asynchronous search experiences of a community of like minded remote searchers to provide improved personalized results. It is based on case-based reasoning [3], an approach which uses previous search experiences of searchers to refine future searches. A case base (c_i) consists of search cases with each search case made up of a specification part ($Spec(c_i)$) and a solution part ($Sol\ (c_i)$) that is represented as follows as shown in Equation 1.

$$c_i = (q_i, (p_1, H \tag{1}$$

The specification part consists of the query q_i and the solution part consists of the number of hits H_j each page p_j has got that belongs to the result set of that particular query. The modified collaborative web search approach can be implemented as a meta-search engine working on a background search engine like Google.

The architecture of the Collaborative Web Search (CWS) is explained in Figure 1. Whenever a searcher submits a query, the query is sent to Google and also to the modified collaborative web search meta-search engine. In collaborative web search meta-search engine, the query is first passed through the pre-processing block. The output query from pre-processing block and the results of the underlying Google search form the input to the Hit data structure which keeps a record of the number of hits a page has got for a particular query. The next processing block does all the computations and presents the promoted list of results R_P to the user.

At the same time, a list of normal results returned by Google is also collected. This forms the standard list R_S. Both promoted list and standard list are merged together and returned to the user as the final result R_{Final}. Normally the promoted results can be shown on top followed by normal Google search results. Otherwise, the promoted results can be shown in one column and standard results in another column.

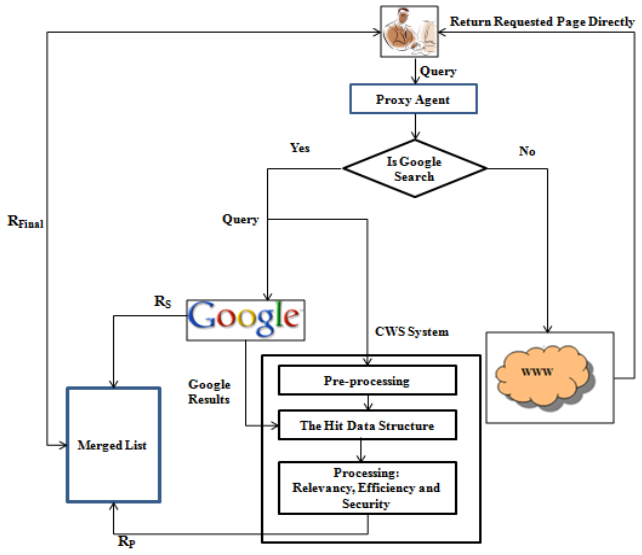

Figure 1. Architecture of Modified Collaborative Web Search

A. *Pre-processing:*

The pre-processing is mainly divided into three stages.

a. ***Stopwords Removal:*** Stopwords are the words which can be filtered out without affecting the results that will be returned [7]. Examples of some of the stopwords include: the, a, is, of etc. Removal of stopwords helps in better finding the similarity between queries. Example, "jaguar photos" and "photos of jaguar" will not be identified as duplicate queries because of the extra 'of' although they are exactly the same queries if used directly. Stopword removal will convert "photos of jaguar" to "photos jaguar" so that the two queries are properly identified as duplicates.

b. ***Stemming:*** Stemming refers to the process of reducing inflected or (sometimes derieved) words to their stem, base or root form. This consists of removal of extra suffix from the words. For example, the use of stemming will convert words such as connect, connecting, connections, connects to their root form connect. Terms with similar stem usually have similar meaning. Hence stemming avoids the duplication of queries in the data structure. Porter Stemming [15] algorithm is used for stemming. It is a widely used stemming algorithm and it is safe enough not to remove a suffix when the stem is too short.

c. ***Finding Synonmyms:*** Users usually type the queries in natural language. In natural language it is quite common that two queries with apparently different word may refer to the same object. For example "picture" and "photo" may be the synonyms of the same query. But they will be identified as different queries. To deal with such queries a lexical database can be used to check if any synonym of the target query is already present in the hit data structure. If present the query is converted to that synonym and results are computed in the processing block. If it's not present, the query is inserted in the data structure and is made available for refinement of further searches.

B. *The Modified Data Structure:*

The hit data structure is used to achieve the collaboration of community and is depicted in Figure 2.

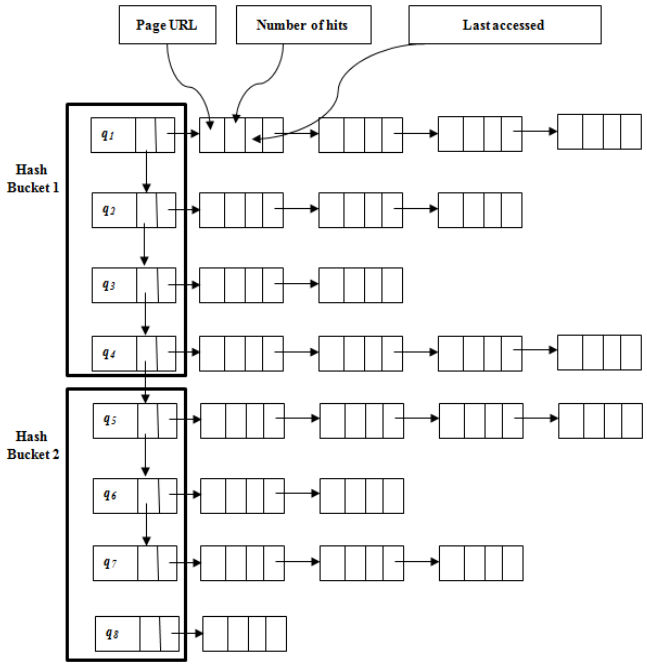

Figure 2. Data Structure used in Modified Collaborative Web Search

In this specially designed modified data structure, the pages are indexed on queries with the pointer from each query leading to a linked list of pages that are associated with that query. For example in the above diagram, the node consisting of query q_1 consists of two pointers. One pointer points to the node containing the next query. The other pointer points to the corresponding linked list of pages associated with that query.

The nodes in the linked list consist of the following four fields.

a. ***The URL of the Page:*** The URL of the page consists of addresses of the pages returned by the underlying search engine.

b. ***The number of hits of the page:*** It refers to the number of times that page has been selected by community members for that particular query.

c. ***Last Accessed Date:*** It helps us to calculate the number of days passed since the last access of the page.

d. ***Pointer to the Next Node:*** This last field is simply a pointer to the next node of the linked list.

Further, the queries are hashed into several buckets. This increases the insertion and retrieval efficiency. Whenever the user submits a query, instead of searching it in the entire linked list of queries to check its presence, we find the hash of the query and see to which bucket it hashes to. Then the search process is carried out only in that bucket. If the query is not present, we can insert that query in that bucket and next time it is available for generating recommendations. Next, we can order the linked list of pages in decreasing order whenever the load on the system is low, based on the number of hits so that pages which have got most number of hits will be located in beginning only and search time will be reduced to a significant extent.

C. *Methodology:*

The first step is finding the similarity. Whenever a new query comes we have to check if that query is already present in the hit data structure. This is done by using Jaccard Similarity Coefficient [6].

For example, if "Pictures of Jaguar" is the target query (q_T) and "Jaguar photo" (q_i) is the one present in hit data structure then, without preprocessing, the similarity (*Sim*) computation using Jaccard similarity coefficient [3] between query q_T and q_i, equals 0.25 as given in Equation 2. The system fails to identify two exact similar queries.

$$Sim(q_T, q_i) = \frac{q_T \cap q_i}{q_T \cup q_i} = \frac{\{Photos\ of\ Jaguar\} \cap \{Jaguar\ Picture\}}{\{Photos\ of\ Jaguar\} \cup \{Jaguar\ Picture\}} \quad (2)$$

The pre-processing steps are as follows. The first step is to remove the stopwords in the tagert query q_T, So the "Pictures of jaguar" will get converted to "Pictures Jaguar". Next, using Porter Stemmer Algorithm [7] we can stem "Pictures Jaguar" to "Picture Jaguar". Finally using a lexical database we can convert "Picture" to "Photo" so that the two queries become similar. Now, using Jaccard correlation coefficient, the similarity (*Sim*) equals:

$$Sim(q_T, q_i) = \frac{q_T \cap q_i}{q_T \cup q_i} = \frac{\{Photo\ Jaguar\} \cap \{Jaguar\}}{\{Photo\ Jaguar\} \cup \{Jaguar\}} \quad (3)$$

The above difference in similarity calculation in Equation 2 and Equation 3 marks the significance of preprocessing steps. Without the preprocessing steps the system may fail to identify two actual duplicate queries.

The relevance (*Rel*) of a page with some target query is calculated as given in Equation 4 where c_i refers to the case base belonging to query q_i and p_j is the page whose relevance we are calculating:

$$Rel\ (p_j, c_i) = \frac{\quad}{\Sigma_{\forall x}} \quad (4)$$

H_j refers to the number of hits that page has got for query. In a community of people having similar interest, pages belonging to that interest category will have higher hit count. n_j refers to the number of days passed since its last access. This creates a bias towards never pages, as the older pages, although they have higher hit count might be no more relevant. Now, the weighted relevance (*WRel*) [6] of page p_i to some new target query q_T can be calculated as given in Equation 5:

$$WRel(p_j, q_1, \ldots, q_n) = \frac{\Sigma_{i=1\ldots n} Rel(p_j, q_i) \cdot S}{\Sigma_{i=1\ldots n} Sim(q_i} \quad (5)$$

The weighted relevance metric rank orders the search results from the community case base and presents the promotion candidates to users for the target query. The promotion candidates are shown to the user in addition to the normal Google search results.

Further, since in this approach it is not possible to identify individual users, malicious users may simply click irrelevant pages to increase their hit counts. To deal with this, instead of users, the check is kept on the pages accessed. If any page is getting accessed far more number of times compared to a threshold in a small time frame, a bias towards that page can be detected which can be an activity of malicious users and a check can be kept on that page. This adds a level of security to the system against malicious activities.

The approach has been implemented on Java platform on test bases. The current implementation is limited to a single community. The approach has helped to deliver reliable relevant personalized results with better recall and relevance score. Also the efficiency compared to the original approach[6] is significantly improved.

V. CONCLUSION

The motivating insight on this research is that there are important features missing from mainstream search engines like Google, Yahoo etc. These engines offer no solution for sharing of the search results between users despite of the fact that there is tremendous potential that can be explored to further refine the quality of search results returned.

The modified approach presented in this paper works on the principle of collaborative web search which allows members of a community to share their search experiences which can benefit other community members. The approach allows like minded people to asynchronously collaborate irrespective of the distance between them and returns improved personalized results. The system has proved to deliver better performance compared to the underlying search engine and the original approach in terms of delivering reliable relevant personalized results and efficiency.

VI. REFERENCES

[1] HeyStaks White Paper, "Social search and search analysis in the discovery economy: an enterprise perspective." August 2013 HeyStaks Technologies Ltd.

[2] A. Pretschner and S. Gauch, "Ontology based personalized search," in Proc. 11th IEEE Int'l Conf. Tools with Artificial Intelligence (ICTAI '99), pp. 391-398, 1999.

[3] Agnar Aamodth and Enric Plaza, "Case-based reasoning: foundational issues, methodological variations and system approaches," AI Communications, Vol. 7 Nr. 1 March 1994, pp 39- 59.

[4] B. M. Sarwar, G. Karypis, J. A. Konstan, and J. Riedl, "Item based collaborative filtering recommendation algorithms," in Proceedings of the 10th International Conference on World Wide Web (WWW '01), pp. 285–295, May 2001.

[5] B. Smyth, E. Balfe, O. Boydell, K. Bradley, P. Briggs, M. Coyle, and J. Freyne, "A live-user evaluation of collaborative web search," in Proceedings of the 19th International Joint Conference on Artificial Intelligence (IJCAI '05). Morgan Kaufmann, 2005, pp. 1419–1424, ediburgh, Scotland.

[6] Barry Smyth, Maurice Coyle and Peter Briggs, "The altrustic seacher," in Proceedings: 12th IEEE International Conference on Computational Science and Engineering: CSE 2009: ol 4.

[7] Chris Buckley and Gerald Salton, "Stop Word List," SMART Information Retrieval System, Cornell University.

[8] Eric J. Glover, Steve Lawrence, Michael D. Gordon, William P. Birmingham, C. Lee Giles, "Recommending web documents based on user preferences," in ACM SIGIR 99 Workshop on Recommender Systems, Berkeley, CA, August, 1999.

[9] J. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," inProceedings of the 14th Conference on Uncertainty in Artificial Intelligence (UAI '98), 1998.

[10] Jay Budzik and Kristian J. Hammond, "User Interactions with everyday applications as context for just-in-time

CONFERENCE PAPER
Two day National Conference on Innovation and Advancement in Computing
Organized by: Department of IT, GITAM UNIVERSITY Hyderabad (A.P.) India
Schedule: 28-29 March 2014

information access ," in Proceedings of the 5th international conference on Intelligent user interfaces Pages 44-51 ACM New York, NY, USA 2000.

[11] Jill Freyne and Barry Smyth, "Cooperating search communities," Springer- Verlag Heidelberg 2006.

[12] Jill Freyne, Barry Smyth, Murice Coyle, Evelyn Balfe and Peter Briggs, "Further experiments on collaborative ranking in community based search," 2004 Kluwer Academic Publishers.

[13] Ji-Rong Wen, Zhicheng Dou, and Ruihua Song, "Personalized web search," in Encyclopedia of Database Systems, Springer-Verlag, New York, USA, September 2009.

[14] Lev Finkelstein, Evgeniy Gabrilovich, Yossi Matias, Ehud Rivlin, Zach Solan, Gadi Wolfman, And Eytan Ruppin, "Placing search in context: the concept revisited," Acm Transactions On Information Systems, Vol. 20, No. 1, January 2002, Pages 116–131.

[15] M. F. Porter, "An example for suffix stripping," Program 14 no. 3, pp 130-137, July 1980.

[16] Meredith Ringel Morris and Eric Horwitz, "Searchtogether: an interface for collaborative web search," UIST 07 Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology.

[17] Meredith Ringel Morris, " A survry of collaborative web search practices," in CHI 2008, pp. 1657-1660.

[18] Page L., Brin S., Motwani R., and Winograd T, "The pagerank citation ranking: bringing order to the web," Technical report, Computer Science Department, Stanford University, 1998.

[19] Peter Briggs and Barry Smyth, "Provenance trust and sharing in peer-to-peer case based web search," Advances in Case-Based Reasoning: 9th European Conference ECCBR 2008 Trier, Germany, September 1-4, 2008, Proceedings (Springer).

[20] R. Burke, "Hybrid recommender systems: survey and experiments,"User Modelling and User-Adapted Interaction, vol. 12, no. 4, pp. 331–370, 2002.

[21] S. Amershi and M. R. Morris, "Cosearch: a system for co-located collaborative web search," in CHI, 2008, pp. 1647–1656.

[22] Xiaoyuan Su and Taghi M. Khoshgoftaar, " A survey of collaborative filtering techniques," Hindawi Publishing Corporation, Advances in Artificial Intelligence Volume 2009, Article ID 421425, 19 pages doi: 10.1155/2009/421425.

[23] Zhicheng Dou, Ruihua Song, Ji- Rong Wen and Xiaojie Yuan, " Evaluating the effectiveness of personalized web search," IEEE Transactions on Knowledge and Data Engineering, vol 21, No. 8, August 2009.