A Re-ranking Strategy for Web Search Personalization

Sanjay Choudhary
Research Scholar, CSE Department
MANIT
Bhopal, India
tosanjaychoudhary@gmail.com

Jaytrilok Choudhary
Assistant Professor, CSE Department
MANIT
Bhopal, India

Vasudev Dehalwar
Associate professor, CSE Department
MANIT
Bhopal, India

Abstract— **Work of relevant information retrieval from search engines is a tedious task in current scenario as there is huge amount of information present on the web. Web Search personalization is the process to filter search results to a particular user based on user's interest and preferences. The judgment to make the search results relevant depends on user and the context of search. So, to judge the user and search context we build a user profile that will be very helpful in obtaining most relevant results .Also we maintain Query log that will help us to solve the ambiguity problem with the queries. Ambiguity refers to the missing weights of certain words in the user profile. In this paper we present an effective filtering strategy that compensates for the ambiguity in a user's profile, by applying re-ranking algorithm. After evaluation the results are effective and show an improvement in precision over approaches that use only a user's profile.**

Keywords: Personalization, Query disambiguation, Re-ranking.

I. INTRODUCTION

Due to the excess information on the WWW, the amount of results returned for a particular keyword search is enormous. This is problematic for the user to scan and navigate the retrieved material to find the web pages satisfying his actual information need. For e.g. two different users may use exactly the same query "Java" to search for different means of information - the Java Island in Indonesia or the Java programming language. Existing IR systems would return a similar set of results for both these users. Building the user's interests and focus into the search process is quite essential for disambiguating the query and providing personalized search results. One way to disambiguate the words in a query is to associate a categorical tag with the query. For example, if the category "software" or the category "travel" is associated with the query "java", then the user's intention becomes clear. By utilizing the selected categories as a context for the query, a search engine is likely to return documents that are more suitable to the user. Current search engines such as Google or Yahoo! have hierarchies of categories to help users to specify his/her categories manually to the query. Unfortunately, such extra effort cannot be expected from the user in a web search scenario. Instead it is preferred to automatically obtain a set of categories for a user query directly by a search engine. However, categories returned from a typical search engine are still independent of a particular user and many of the returned document results could belong to categories that may not reflect the intention of the searcher. This demands further personalization of the search results.

We use the respective categories for the query along with the corresponding clicked documents in the learning of a user profile for the user. The user profile is represented as a matrix containing the pairs (term, category) and their corresponding weights. Machine learning algorithms are used to automatically learn these term weights in the matrix. Each element represents how important the term is when the user is searching for a query of the given category. Re-ranking of the search results based on the user profile thus built, has shown improvement in performance. Though category helps to disambiguate the query, it adds another extra dimension to the user profile. This typically brings in ambiguity in the user profile, which was observed in our case. Ambiguity refers to

the missing weights of certain words in the user profile. In this paper, we present an effective re-ranking solution that compensates for the ambiguity in a user's profile, by collaborative filtering algorithms.

II. RELATED WORK

The tremendous growth and popularity of the World Wide Web has resulted in a enormous amount of information sources on the Internet, creating a scenario where the answers to information needs of the users are available online somewhere in some format; but in order to find the appropriate information users need to scan through endless list of digital data. Different types of users explore the Web in various ways according to their requirements and experiences; some users, for instance, may survey an area of knowledge to get a general understanding on it, while others to look for specific information. In either of the cases, they need to access and analyze all the documents available and this process is time consuming. For these reasons, they normally tend to compromise themselves with the information they have received [1]. Personalized web content [2] is one of the proposed solutions to solve this problem. Most part of information is present in the form of unstructured free text, written in natural languages. Examples are blogs, forum, corporate memos, research reports, emails, blogs and historical documents [3] .According to recent studies more than 80% of queries submitted by users to search engines are estimated informational in nature. This means that most of them could be answered properly by providing structured and normalized form of information, like to key notes of entities, price lists of items for sale, document summaries. The purpose of Information extraction (IE) [4] is to structure the possible unstructured text; in other words, IE is the process of populating a template of structured information starting from unstructured or loosely formatted text, which can be given directly to user or can be stored in a database for further processing. Reference [5] suggested improving retrieval efficiency by tracking the user and exploring his/her logs. The authors reported that their algorithm dramatically improved the result's efficiency. They investigated the user's log files in the search engine and used them in the subsequent queries. This method directs the search engine toward common information in documents for each user.

Reference [6] investigated the possibility to find a web page relevant to a reference web page. Although the objective of the project is quite similar to this paper, it was implemented using a totally different strategy. The authors used the reference page only to represent the knowledge-based system. Reference [7] in proposed a similar method to the one provided in this paper. The authors provided a personalized web search for improving retrieval effectiveness. They have implemented a machine learning algorithm to capture the user interests. Every time the user connects to a URL, the system keeps track of that URL and categorizes it. This process improved the overall system performance as every URL is reflected on the subsequent query.

III. PROBLEM

To improve the retrieval effectiveness for personalizing the web search is the main problem. Our strategy includes three steps. The first step is to map a user query to a set of links. The second step is to utilize both the query and its context to retrieve Web pages using ontology [8].

In order to do the first step, ODP is used as a resource. A hierarchical model approach is used to represent a user's search history and describe how a user's search history can be collected without his/her direct involvement. The user submits a query to the search engine. The search engine produces set of results composed with the relevant and irrelevant page collections.

Table 1. Sample Category and Term matrix

Category / Term	**MS Corp.**	**MSDN**	**Windows**	**MS Office**	**MS careers**
Microsoft Corp.	1	0.4	0.5	0	0.5
Microsoft Software	0	0	1	0.6	0

Table2. Sample Document and Category matrix

Doc/ Category	**MS Corp**	**MSDN**	**Windows**	**MS Office**	**MS careers**
D1	1	0	0	0	0
D2	0.5	0	0	0	0.5
D3	0	0.5	1	0	0.5
D4	0	0	0	1	0

Table3. Sample Document and Term matrix

Doc / Term	Microsoft Corporation	Microsoft Software
D1	1	1
D2	0	0
D3	1	0
D4	0	0

The relevant or irrelevant page identification .is a complex task to the user. Anyway, the presence of unwanted pages in the result set would force him or her to perform a post processing on retrieved information to discard unneeded ones.

With respect to other ranking strategies for the Semantic Web [9], our approach only relies on the knowledge of the user query, the Web pages to be ranked, and the underlying ontology. Hence, it allows us to effectively manage the search space and to reduce the complexity associated with the ranking task.

IV. PROPOSED ALGORITHM

The Hyperlink Induced Topic Search (HITS) algorithm is used for the page ranking process [10]. The results are ranked and irrelevant pages are removed from the result.

Table4. Sample Category and Link table

Category/Link	Microsoft Corporation
Microsoft Corporation	en.wikipedia.org/wiki/**Microsoft** www.**microsoft.com**/India www.**microsoft.com**/en/in/default.aspx **bhashaindia.com** in.finance.yahoo.com/q?s=MSFT timesofindia.indiatimes.com/topic/**Microsoft-Corporation** iplextra.indiatimes.com/topic/**Microsoft_Corporation** das**.microsoft.**com/activate/en-us/default.asp office**.microsoft.**com/en-in support**.microsoft.**com/kb/91728

Table5. Sample Category and User matrix

Cat/ User	MS Corp	MSDN	Windows	MS Office	MS careers
MS Corp	0	1	0	0	1
MS Soft	0	0	0	0	0

Links relevant to the Microsoft Corporation and Microsoft software is prioritized based on term weight. Every user is mapped with category and based on their search history weight is assigned. Based on the user's preference the relevant links will be prioritized and the irrelevant links will be omitted. In the above table user is more interested on MSDN and Microsoft careers so other links will be hidden for the users. Rocchio [12] is originally a relevance feedback method. We use a simple version of Rocchio adopted in text categorization

$$M(i.j) = Max \sum_{k=1}^{m} DT(k,j) * DC(k.i) \qquad (1)$$

where M is the matrix representing the user profile, N is the number of documents that are related to the ith category, m is the number of documents in DT, DT(k,j) is the weight of the jth term in the kth document, DC(k,i) is a binary value denoting whether the kth document is related to the ith category .clearly, M(i,j) is the max weight of the jth term in all documents that are related to the ith category and documents that are not related to the category are not contributing to M(i,j). We call it as MaxRocchio method .Based on the category term weight, Category link will be prioritized. To include personalization for every user, category term weight will be calculated.

$$MU(i,j) = \sum_{u=1}^{n} \sum_{k=1}^{m} DT(k,j) * DC(k.i) \qquad (2)$$

Interested terms links will be mapped with the user and will be displayed

V. RESULT AND DISCUSSION

A. Measure of Web Page Retrieval

The measure of effectiveness is essentially the ―Precision at 11 standard recall levels as used in TREC evaluation [11]. It is briefly described as follows:

A.) For each query, for each list of retrieved documents up to the top 20 documents, all relevant documents are identified. (In practice, a number higher than 20 may be desirable. However, we have a limited amount of human resources to perform manual judgment of relevant documents. Furthermore, most users in the Web environment examine no more than 20 documents per query.)

B.) The combination of all relevant documents in all these lists is assumed to be the set of relevant documents of the query.

C.) For each value of recall (percentage of relevant documents retrieved) from all the recall points {0:0; 0:1; . . .; 1:0}, the precision (the number of relevant document retrieved divided by the number of retrieved documents) is computed.

D.) At the end, the precision, averaged over all recall points, is computed. For each of the data set and for each mode of retrieval, we will obtain a single precision value by averaging the precision values for all queries

E.) Now, we examine the efficiency of our method by calculating the average times for processing a query in seconds. Different times are as follows:

a) The time to map the user query to a set of categories,

b) The time for the search engine, Yahoo Directory, to retrieve the documents,

c) The time for our system to extract lists of documents from the search engine result pages, and

d) The time to map user interested links from retrieved results

Thus, the portion of our algorithm which consists of step a and d is efficient.

VI. CONCLUSION

To personalize search results, we had a strategy that will first learn a user profile from his "click through data", collected from a real world search engine. This user profile is then used in a re-ranking phase to personalize the search results. We also used query and its category information to in learning the user profile. Category information helps to disambiguate the query and focus on the information need. We propose an effective re-ranking strategy that compensates for the ambiguity in a user's profile, using collaborative filtering algorithms. We evaluate our approach using standard information retrieval metrics, to show an improvement in performance over earlier re-ranking strategies based on only user profile.

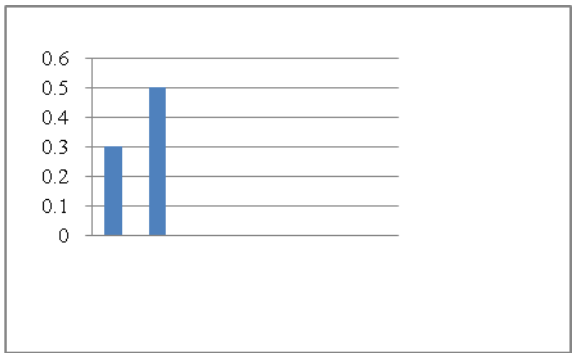

Figure.1: Comparison between precision before and after our approach (0.3 before and 0.5 after our approach)

The ranking scheme is improved with the content and hyperlink in the web pages. The user can easily identify the relevant pages. The ranking scheme produces better results than other ranking.

REFERENCES

[1] S. Gauch, M. Speretta, A. Chandramouli, and A. Micarelli, User profiles for personalized information access, in The Adaptive Web

[2] Lecture notes in Computer Science, P. rusilovsky, A. Kobsa, and W. Nejdl, Eds., vol. 4321. Springer, 2007, pp. 54–89.

[3] P. Brusilovsky and C. Tasso, ―Preface to special issue on user modeling for web information retrieval, User Model. User- Adapt. Interact. vol. 14, no. 2-3, pp. 147– 157, 2004.

[4] A. McCallum, ―Information extraction: distilling structured data from unstructured text, ACM Queue, vol. 3, no. 9, pp. 48–57, 2005.

[5] Hang Cui; Ji-Rong Wen; Jian-Yun Nie; Wei-YingMa, ―Query expansion by mining user logs, IEEE Transactions on Knowledge and Data Engineering, page(s): 829- 839, July-Aug. 2003.

[6] Taher Haveliwala, Aristides Gionis , Dan Klein, and Piotr Indyk. ―Evaluating strategies for similarity search on the web, In Proceedings of the Eleventh International World Wide Web Conference, May 2002.

[7] Fang Liu; Yu, C.; Weiywe Meng. ―Personalized web search for improving retrieval effectiveness, IEEE Transaction on knowledge and data engineering, Volume: 16, Issue: 1, page 28-40, Jan. 2004.

[8] M. Eirinaki, D. Mavroeidis, G. Tsatsaronis, and M. Vazirgiannis. Introduction to semantics in web personalization: the role of ontologies, EWMF/KDO5, 2006.

[9] A. Sieg, B. Mobasher and R. Burke. Web search Personalization with Ontological user Profiles, CIKM'07, 2007.

[10] M. S. Aktas, M. A. Nacar, and F. Menczer. Using hyperlink features to personalize web search, WWW, 2005.

[11] E.M. Voorhees and D. Harman, eds., ―Common Evaluation Measures, Proc. Text Retrieval Conf.(TREC-10), p.A-14 ,2001.

[12] Saboori, F; Bashiri, H; and Oroumchian, Farhad: Assessment of query reweighing, by rocchio method in farsi information retrieval 2008.