Supporting Privacy Protection in Personalized Web Search

Sarika N. Zaware, Deepali Shah, Taahir Hamgi, Payal Khinvasara, Rasika Tribhuwan
Department of Computer Engineering, AISSMS IOIT, Pune

Abstract— Personalized search engine is a search engine that helps users to pick up the useful information for them quickly according to their interest, which are stored in the database. Personalized search engine can sort the results according to users' interest, the results that user likes will beat the top of the results.Unnecessary exposure will be avoided and relevant results will be obtained. When a query is fired, the results provided by the search engine are filtered using ODP (Open Directory Project) operations. When a similar query is fired again, more precise results will be displayed to user. Thus Personalized web search has been efficiently carried out. It is a good measure to use Vector Space Model to help us implement the personalization. We use Vector Space Model to model the user interest and the result, then we use cosine angel to calculate the similarity of these interest. This paper describes the design and implementation of this system by using result and user modelling.

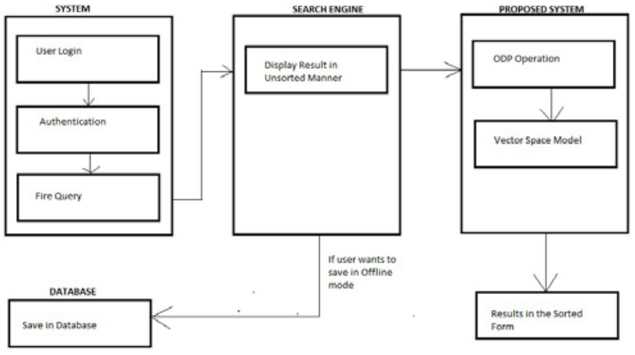
Keywords— Privacy Protection, personalized web search, utility, ODP, profile.

I. INTRODUCTION

Advent of the information age, the Internet can enable people to access information more easily. On the other hand, with today's information age knowledge explosion, the search engines are more important in our life. Since the search engine can get more information from many sources, there are pretty of information that users don't care about. This advantage turns to disadvantage. It makes user to use more time to deal with the information they are not interested in. Against the background, personalized search engine is one way to solve the problem.

The mean of personalization is search engine can help users to filter the useful information for them by using user's interest. Search engine will pick the users' interest at the top of results, so it is very convenient for users to pick useful information. In this paper we will introduce the design and implementation of personalized search engine. We model the results and users' interest according to the Vector Space Model. The profile-based PWS has demonstrated more effectiveness in improving the quality of web search with increasing usage of personal and behavior information to profile its users.

II. SYSTEM MODEL



BLOCK DIAGRAM OF PROPOSED SYSTEM

As we can see in this block diagram, there are 2 main components involved in the working of our personalized web search. The prior is the System which is further extended as the proposed system where the re-ranking of the pages obtained from the search engine i.e 2nd component is done.

In the initial stage, the user is asked to log in into the system. The autentication is done and user can now fire a query. This query is forwarded to the search engine i.e Google Search in our model.

Once the results are obtained from the search engine they are categorized using ODP operations, which help us to determine the user interests also. Once the results are obtained we re-rank the pages for the next session of the user.

For re-ranking the pages we use the vector space model/algorithm.The Vector Space model will be discussed further in detail.

III. PREVIOUS WORK

The question arises about the need of our system when there are already some huge search engines available like Google, Yahoo, etc.

But the thing to be noted about these search engines is that their results are common to all the users irrespective of their interests, area of work, or their behavior.

When a user enters a same/similar query in any of these search engines, the result obtained is same as before irrespective of previous choice.

The order of the pages/links displayed is never changed based on the user profile or interests. Suppose that a user is interested in a link which is displayed on the 3rd of 4th page, and next time if the user wants to access the same link, still he would have to browse though the 3rd of 4th page.

The ranking of the pages in these search engines is basically dependant on popularity based page ranking and advertisement revenues. This makes it very difficult for the user to get prioritized results.

Now, we can overcome these drawbacks of the existing system through our proposed system

In our search engine, we provide prioritized results to the user based on this profile and interests and his search behavior.

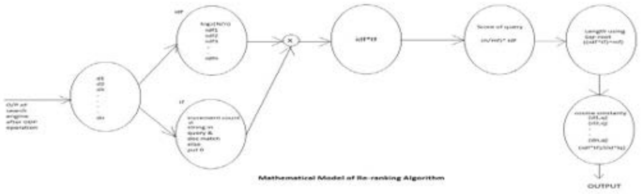
1. Supporting privacy protection in personalized web search. We came to know that profile based methods can be potentially effective for almost all sort of queries. PWS has demonstrated more effectiveness in improving the quality of web search.[7]
2. Personalized Search engine design and Implementation. We got to know that vector space model is an efficient for re-ranking the web search results. It discards the unwanted results and store or rank results according to user interests and behavior, with the help of ODP.[5]
3. Automatic identification of user interest for personalized search. Here it shows how search engine can learn a user’s preference to personalize search results.[6]
4. Personalized concept-Based clustering of search Engine Queries. In this paper we proposed new personalized query for individual users based on their conceptual profiles.[2]

Thus we can conclude from our literature survey that the existing systems provide results based on the clicks/hits made by other users and they may sometime be irrelevant to our query. But, using our proposed system we are eliminating this drawback and providing personalized search results to every user using ODP and vector space model.

IV. PROPOSED METHODOLOGY

We are using vector space algorithm /Model to personalize the web search. Vector space algorithm for page re-ranking provides prioritized results.

VECTOR SPACE MODEL:



Mathematical Model of Re-ranking Algorithm

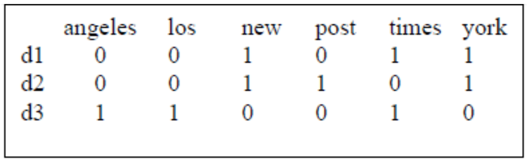
This is a simplified example of the vector space retrieval model. Consider a very small collection C that consists the following three documents:

d1: “new york times”
d2: “new york post”
d3: “los angeles times”

Some terms appear in two documents, some appear only in one document. The total number of documents is *N*=3. Therefore, the *idf* values for the terms are:

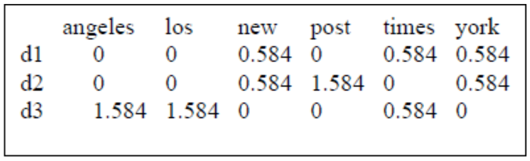
angles log_2(3/1)=1.584
los log_2(3/1)=1.584
new log_2(3/2)=0.584
post log_2(3/1)=1.584
times log_2(3/2)=0.584
york log_2(3/2)=0.584

For all the documents, we calculate the *tf* scores for all the terms in C. We assume the words in the vectors are ordered alphabetically.



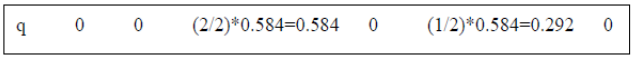
	angeles	los	new	post	times	york
d1	0	0	1	0	1	1
d2	0	0	1	1	0	1
d3	1	1	0	0	1	0

Now we multiply the *tf* scores by the *idf* values of each term, obtaining the following matrix of documents-by-terms: (All the terms appeared only once in each document in our small collection, so the maximum value for normalization is 1.)



	angeles	los	new	post	times	york
d1	0	0	0.584	0	0.584	0.584
d2	0	0	0.584	1.584	0	0.584
d3	1.584	1.584	0	0	0.584	0

Given the following query: “new new times”, we calculate the *tf-idf* vector for the query, and compute the score of each document in C relative to this query, using the cosine similarity measure. When computing the *tf-idf* values for the query terms we divide the frequency by the maximum frequency (2) and multiply with the *idf* values.



q	0	0	(2/2)*0.584=0.584	0	(1/2)*0.584=0.292	0

We calculate the length of each document and of the query:

Length of d1
= sqrt(0.584^2+0.584^2+0.584^2)=1.011
Length of d2
= sqrt(0.584^2+1.584^2+0.584^2)=1.786
Length of d3
= sqrt(1.584^2+1.584^2+0.584^2)=2.316
Length of q
= sqrt(0.584^2+0.292^2)=0.652
Then the similarity values are:
cosSim(d1,q)
= (0*0+0*0+0.584*0.584+0*0+0.584*0.292+0.584*0)
/ (1.011*0.652) = 0.776
cosSim(d2,q)
= (0*0+0*0+0.584*0.584+1.584*0+0*0.292+0.584*0)
/ (1.786*0.652) = 0.292
cosSim(d3,q)
= (1.584*0+1.584*0+0*0.584+0*0+0.584*0.292+0*0)
/ (2.316*0.652) = 0.112

According to the similarity values, the final order in which the documents are presented as result the query will be: d1, d2, d3.

Analysis:

Time Complexity = O(mn)

Where,
n = Number of documents in database.
m = Number of web pages.

The algorithm for document similarity is NP-complete

V. CONCLUSION

Thus in our paper we are providing prioritized and personalized results to the user using vector space algorithm and ODP operations. Hence, the accuracy and quality of the search is improved. Which leads to better search results is less time.

VI. FUTURE SCOPE

As our application is platform independent, we can create a mobile search computing application.

REFERENCES

[1] Z. Dou, R. Song, and J.-R. Wen, “A Large-Scale Evaluation and Analysis of Personalized Search Strategies,” Proc. Int’l Conf. World Wide Web (WWW), pp. 581-590, 2007.

[2] J. Teevan, S.T. Dumais, and E. Horvitz, “Personalized concept-Based clustering of search Engine Queries” Proc. 28th Ann. Int’l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR), pp. 449-456, 2005.

[3] M. Spertta and S. Gach, “Personalizing Search Based on User Search Histories,” Proc. IEEE/WIC/ACM Int’l Conf. Web Intelligence (WI), 2005.

[4] B. Tan, X. Shen, and C. Zhai, “Mining Long-Term Search History to Improve Search Accuracy,” Proc. ACM SIGKDD Int’l Conf. Knowledge Discovery and Data Mining (KDD), 2006.

[5] Personalised Search engine design and Implementation.Jiandong Cao,Yang Tang,Binbin Lou,2010,IEEE.

[6] Automatic identification of user interest for personalized search,Feng Qiu,Junghoo Cho, MAY 2006,uk.

[7] Supporting privacy protection in personalized web search, Lidan Shou, He Bai, Ke Chan, And Gang Chen, 2014.