# Web Search Result Personalization using Web Mining

Kavita D. Satokar
B.V.U.C.O.E
Katraj,Pune

Prof..S.Z.Gawali
Assistant Professor
B.V.U.C.O.E
Katraj,Pune

## ABSTRACT
The information on the web is growing dramatically. The users have to spend lots of time on the web finding the information they are interested in. Today, he traditional search engines do not give users enough personalized help but provide the user with lots of irrelevant information. In this paper, we present a personalize Web search system, which can helps users to get the relevant web pages based on their selection from the domain list. Thus, users can obtain a set of interested domains and the web pages from the system. The system is based on features extracted from hyperlinks, such as anchor terms or URL tokens, user interest domains and past search results. Our methodology uses an innovative weighted URL Rank algorithm based on user interested domains and user query.

## General Terms
Web Mining, Data Mining, Web Personalization

## Keywords
personalization, recommendation, interested domains, collaborative filtering.

## 1. INTRODUCTION
The explosive growth of documents in the Web makes it difficult to determine the most relevant documents for a particular user, given a general query. Recent search engines rank pages by combining traditional information retrieval techniques based on page content, such as the word vector space [3, 4], with link analysis techniques based on the hypertext structure of the Web [5, 6].Traditional search engine has dealt with searching information on the web to a large extent, but it also has some problems at present. [2]

- The web information has enlarged from quantity to types, showing the trend of exponential growth, so the search engine cannot index all the pages;
- The web information has changed dynamically, so the search engine can not be sure to update in time;
- Traditional search engine can not meet the increasing need day by day that people want personal service for information retrieve;
- Search engine requires hardware owning more storage capacities, even hundreds of GB, and more servers.

Besides the above stated problem a recent research has shown that only 13% of search engines show personalization characteristics. Hence web personalization [1] is one of the promising approaches to tackle this problem by adapting the content and structure of websites to the needs of the users by taking advantage of the knowledge acquired from the analysis of the users' access behaviors. One research area that has recently contributed greatly to this problem is web mining. Web mining aims to discover useful information or knowledge from the Web hyperlink structure, page content and usage log. There are roughly three knowledge discovery domains that pertain to web mining: Web Content Mining, Web Structure Mining, and Web Usage Mining. Web content mining is the process of extracting knowledge from the content of documents or their descriptions. Web document text mining, resource discovery based on concepts indexing or agent based technology may also fall in this category. Web structure mining is the process of inferring knowledge from the World Wide Web organization and links between references and referents in the Web. Finally, web usage mining, also known as Web Log Mining, is the process of extracting interesting patterns in web access logs.

A key part of the personalization process is the generation of user models. It is purely based on observed patterns, and resulting probabilities. Commonly used user models are still rather simplistic, representing the user as a vector of ratings or using a set of keywords. Even where more multi- dimensional information has been available, such as when collecting implicit measures of interest, the data has traditionally been mapped onto a single dimension; in the form of ratings .In particular profiles commonly used today lack in their ability to model user context and dynamics. Users rate different items for different reasons and under different contexts. The user interests and needs change with time. Identifying these changes and adapting to them is a key goal of personalization. We suggest that the personalization process be taken to a new level, a level where the user does not to be actively involved with the personalization process. All that the user needs to do is to have an active profile file and when the user logs onto a web site, the browser checks for that profile file as it checks for the cookies. The profile file describes the user's interest and the levels at which the user wants a particular personalizable feature. Since the profile file is in a standardized format, the web sites would be able to provide the content according to the profile file. This would enhance the user's personalization process without their active involvement.

Classic information retrieval usually used ranking algorithms based solely on the words in the documents. One such algorithm is the *vector space model introduced* by Salton and associates[7]. It onsides a high-dimensional vector space with one dimension per term. Each document or query is represented as a *term vector* in this vector space. Entries of terms occurring in the document are positive, and entries of terms not occurring in the document

are zero. More specifically, the entry of the term is usually a function that increases with the frequency of the term within the document and decreases with the number of documents in the collection containing the term. The idea is that the more documents the term appears in, the less characteristic the term is for the document, and the more often the term appears in the document, the more characteristic the term is for the document.

The present paper proposes a slightly different ranking based on URL. The ranking is *query-dependent* . *The proposed* algorithm assigns a score that measures the quality and relevance of a selected set of pages depending on their URL to a given user query. The basic idea is to build a query-specific two dimensional vector table, called a relevance table, and perform URL analysis on it. Ideally, this table will contain only URLs on the query topic. We propose the following approach for building a relevance table:

1. A *start set* of documents matching the query is fetched from a search engine (say, the first 1000 matches).
2. The start set is augmented by its weight, which is assigned depending on the occurrence of tokens .
3. Each URL is again assigned weights according to user domains ,favorites and expertise which will be obtained from user profile.
4. Now all the records are again ranked in descending order of weights.
5. Select records whose weight matches the weight of the user query.

## 2. PROPOSED ARCHITECTURE

The Web personalization process include (a) The collection of Web data, (b) The modeling and categorization of these data (preprocessing phase), (c) The analysis of the collected data, and (d) The determination of the actions that should be performed. When a user sends a query to a search engine, the search engine returns the URLs of documents matching all or one of the terms, depending on both the query operator and the algorithm used by the search engine. *Ranking* is the process of ordering the returned documents in decreasing order of relevance, that is, so that the "best" answers are on the top. When the user enters the query ,the query is first analyzed .The Query is given as input to the semantic search algorithm for separation of nouns ,verbs, adjectives and negations and assigning weights (3,2,1,-1) respectively. The processed data is then given to the personalized URL Rank algorithm for personalizing the results according to the user domain, interest and need. The sorted results are those results in which the user is interested. The personalization can be enhanced by categorizing the results according to the types.

Thus after building the knowledge base, the system can give use recommendation based on the similarity of the user interested domain and the user query . The recommendation procedure of the System has two steps:

1. The system gives user a list of interested domains .Detect user's current interested domain.

2 Based on user's current interested domain ,past search history and combined his or her profile, the system will give him or her set of URLs with ranking scores.

In this way, the system could help the user to retrieve his or her potential interested domains. Besides, a user can change his or her current interested domain by clicking the interested domain list on the same page but with more convenience. In the beginning, if the user does not have a profile in the database, the system displays the user available domains, and then keeps a track of the user's selections .The user's selections is used to construct a table that uses URL weight calculation. The current interested domains recommendation is based on last selections. The figure1 shows the complete process.

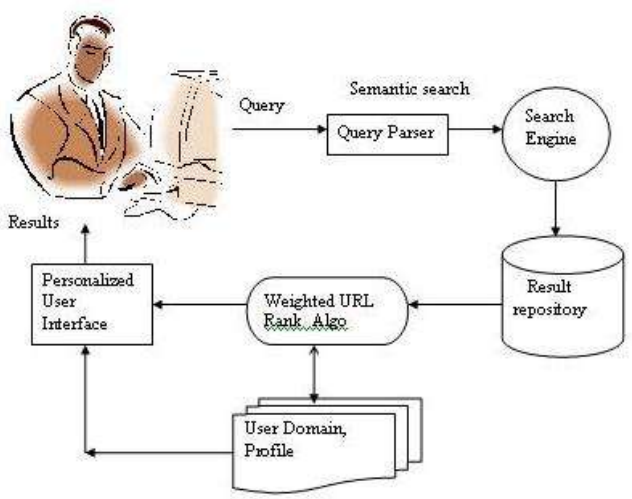


Figure 1: Proposed Architecture

### 2.1 The Experimental Setup

Every user has an associated interested area. The user logs in and posts a query to the system. The query is parsed and nouns , verbs , adjectives and negation are identified and separated using the semantic search algorithm. The query is then given to the search engine to identify related URLs. The extracted page is parsed for given nouns, verbs, adjectives and negations. The URLRank algorithm is used for ranking the identified URLs and assigning corresponding weights (3,2,1,-1) to it. The weights are added up to find the weight of URL. We suppose I = {I |"i"is an interested domain} is a set of all domains in which the user is interested . I' is a subset of I. I0 is a set of interested domains. For each interested domains we assign a corresponding weight to it. If an interested domain appears in I0, the corresponding value of this interested domain is added as weight. Otherwise, it should be zero. U = {u|"u"is a Url } is a set of all urls satisfying the users query with weight. U0 is a subset U. U0 is a recommendation ranked list of URLs, which is based on the user's selection in the I. In other words, for each interested domain i in I, there is a list of URLs U0 corresponds to it. The URLs are again ranked according to favorites and user profile .The URLs are thus arranged in descending order. Thus the URLs appearing higher in the order are those in which the user is most interested. The following flowchart illustrates the complete the process:

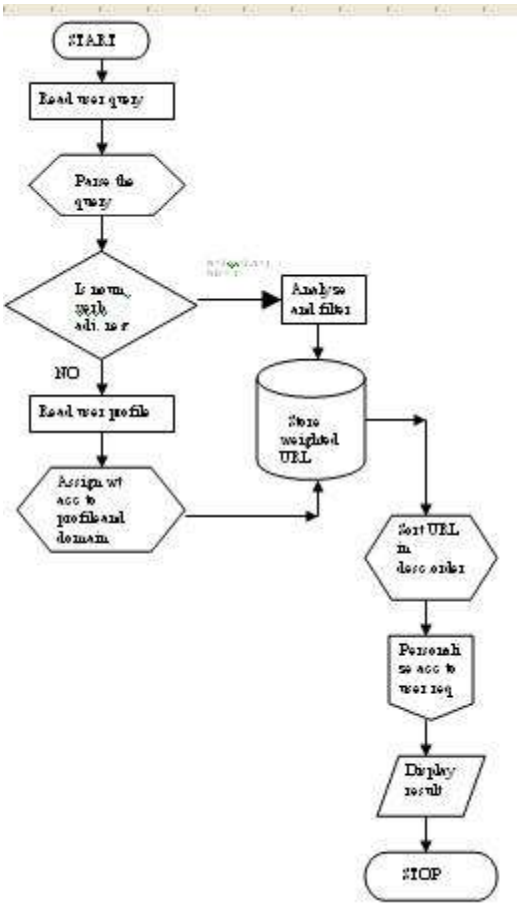


Figure 2: Flow chart of process

## 3. RESULTS

We conducted a user study to compare the performance of the general ranking methods use by existing search engines and weighted (personalized) URLRank. We asked each volunteer to use our personalized search facility after they input their domain profiles into our system. There were 10 human subjects who contributed to our user study with a total of 40 queries. Volunteers were expected to select relevant URLs satisfying their choice of preferences. After submitting a query, a volunteer was shown a single screen with the search results from the proposed personalize search and general search .For each query, the top 10 results from each ranking method were shown to the volunteer. As an example, suppose that JAVA returns at least 30 results satisfying a query. The user URL selection was studied and feedback regarding relevance was taken.

The personalization accuracy was found to be 74%; the random search accuracy is 72.8 %. The average of personalization accuracy is 73.4%. Because the interested domains personalization is done considering the user selected domain, the accuracy is higher than the random recommendation in our experiment. Fig. 3 is a comparison of the interested domains personalization accuracy based on random selection and based on our personalization method.

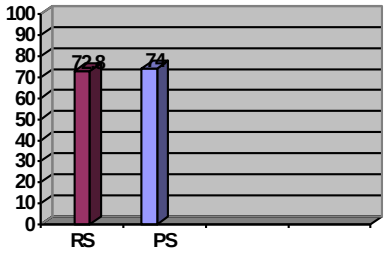


Figure 3: Interested domains personalization accuracy

The URL personalization accuracy based on the interested domains selection is 63.9%; and the URL personalization accuracy without the interested domains selection assistance is 39.9 %. From this result, we can see that the interested domains recommendation help the system to filter lots of URLs that the user might not be interested in. Moreover, the system could focus on the domains that users are interested in to select the relevant URL

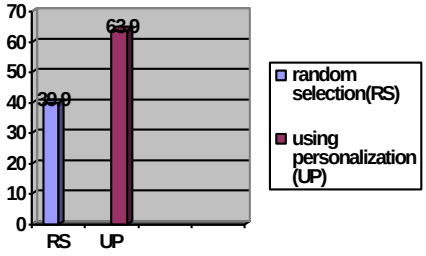


Figure 4: URL Personalization accuracy

## 4. CONCLUSION

In this paper, we present a web personalization system for web search, which not only gives user a set of personalized pages, but also gives user a list of domains the user may be interested in. Thus, user can switch to different interests when he or she is surfing on the web for information. Besides, the system focuses on the domains that the user is interested in, and won't waste lots of time on searching the information in the irrelevant domains. Moreover, the recommendation won't be affected by the irrelevant domains, and the accuracy of the recommendation is increased.

In our experiment, the number of humans involved was very small. Only 10 users' data were used as training data, and 2 users were involved for the testing purpose. Although, there was no other people involved in during the testing phrase, the results still can be somehow biased by their personal behavior. Later, the system should be published on the web and tested by more people. Thus, the interested domains and the URL recommendation can be given when the user is using the proposed tool.

## 5. REFERENCES

[1] M. Eirinaki, M. Vazirgiannis, "Web mining for Web personalization", ACM Transactions on Internet Technology, 3(1), 2003, pp. 1–27.

[2] Ouyang Liubo, Li Xueyong, Li Guohui,Wamg Xin, "A Survey of Web Spiders Searching Strategies of Topic-specific Search Engine," Computer Engineering, vol. 13, pp. 32-33, 46, 2004.

[3] van Rijsbergen, C.: Information Retrieval. Butterworths, London (1979) Second edition.

[4] Salton, G., McGill, M.: An Introduction to Modern Information Retrieval. McGraw-Hill, New York, NY (1983)

[5] Brin, S., Page, L.: The anatomy of a large-scale hypertextual Web search engine. Computer Networks 30 (1998) 107–117

[6] Kleinberg, J.: Authoritative sources in a hyperlinked environment. Journal of the ACM 46 (1999) 604–632

[7] G. Salton et al., "The SMART System—Experiments in Automatic Document Processing," Prentice-Hall, Englewood Cliffs, N.J., 1971.

[8] Hidalgo-Herrero, M., Rodriguez, I., Rubio, F.: Testing learning strategies. In: ICCI '05: Proceedings of the Fourth IEEE International Conference on Cognitive Informatics, Washington, DC, USA, IEEE Computer Society (2005) 212– 221