Personalized Ranking of Search Results with Implicitly Learned User Interest Hierarchies

Hyoung-rae Kim and Philip K. Chan
Computer Sciences Department Technical Report CS-2005-11
Florida Institute of Technology
Melbourne, FL. 32901, USA
+1-321-674-7280
hkim@cs.fit.edu, pkc@cs.fit.edu

ABSTRACT
Web search engines are usually designed to serve all users, without considering the interests of individual users. Personalized web search incorporates an individual user's interests when deciding relevant results to return. We propose to learn a user profile, called a user interest hierarchy (UIH), from web pages that are of interest to the user. The user's interest in web pages will be determined implicitly, without directly asking the user. Using the implicitly learned UIH, we study methods that (re)rank the results from a search engine. Experimental results indicate that our personalized ranking methods, when used with a popular search engine, can yield more relevant web pages for individual users.

Categories and Subject Descriptors
H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval – *retrieval models, selection process, information filtering, data mining.*

General Terms
algorithms, experimentation.

Keywords
personalized web search results, implicit user interest hierarchy, scoring function, user profile, contents based method.

1. INTRODUCTION
Web personalization adapts the information or services provided by a web site to the needs of a user. Web personalization is used mainly in four categories: predicting web navigation, assisting personalization information, personalizing content, and personalizing search results. Predicting web navigation anticipates future requests or provides guidance to client. If a web browser or web server can correctly anticipate the next page that will be visited, the latency of the next request will be greatly reduced [11,20,33,4,10,13,38,39]. Assisting personalization information helps a user organize his or her own information and increases the usability of the Web [27,24]. Personalizing content focuses on personalizing individual pages, site-sessions (e.g., adding shortcut), or entire browsing sessions [1]. Personalized web search results provide customized results depending on each user's interests [19,18,25,2,17,30,3]. In this work, we focus on personalizing web search by ordering search engine results based on the interests of each individual user, which can greatly aid the search through massive amounts of data on the internet.

There are two main techniques for performing web personalization: collaborative filtering [4,10] and user profiles [29,34]. Collaborative filtering uses information from many different users to make recommendations. Collaborative filtering assumes that people have common interests, and would suggest web pages that are the most popular. Disadvantages of this method are that it cannot predict whether a user will like a new page, and it requires a large amount of data from many users to determine what pages are the most popular. Obtaining data on the web pages visited by many different users is often difficult (or illegal) to collect in many application domains. In contrast, user profiles require the web page history of only a single user. There are two techniques for building a user profile: explicit and implicit. The explicit approach has major disadvantages. It takes time and effort for a user to specify his or her own interests, and the user's interests could change significantly over time. Alternatively, an implicit approach can identify a user's interests by inference, and can automatically adapt to changing or short-term interests.

In this paper, we propose a method to personalize web search by ranking the pages returned from a search engine. Each page's ranking is determined by using the individual's implicitly-learned user profile. Pages are ranked based on their "score," where higher scores are considered to be more interesting to the user after comparing the text of the page to the user's profile. For example, if a user searches for "Australia" and is interested in "travel," then links related to "travel" will be scored higher; but if a user is interested in "universities," then pages related to "universities" will be scored higher. We wish to devise a scoring function that is able reorder the results from Google [14], based on a user's implicitly learned interests, such that web pages that the user is most interested in appear at the top of the page. A User Interest Hierarchy (UIH) is built from a set of interesting web page using a divisive hierarchical clustering algorithm. A UIH organizes a user's interests from general to specific. The UIH can be used to build a scoring function for personalizing web search engines or e-commerce sites [5].

While using a search engine, people find what they want. Often times they also find web pages they want to visit next time again. We define *interesting* web pages and *potential interesting* web pages. The definition of *interest* is whether a user found what they want; the definition of *potential interest* is whether a web page will be interesting to a user in the future.

Our contributions are:

- We introduce personalized ranking methods (WS and US) that utilize an implicitly learned user profile (UIH);
- We identify four characteristics for terms that match the user profile and provide a probabilistic measure for each characteristic;
- Our experimental results indicate that WS method can achieve higher precision than Google for Top 10, 15 and 20 web pages that are relevant to the user search query;
- The WS method can also yield higher precision than Google for Top 1, 5, 10, 15 and 20 web pages that are *potentially interesting* to the user;
- When incorporating the (*public*) ranking from the search engine, we found that equal weights for the public and personalized ranking can result in higher precision.

The rest of this paper is organized as follows: Section 2 presents related work regarding personalized search results and the use of bookmarks; Section 3 details our approach to reorder search results; Section 4 provides a detailed description of our user-interest scoring methods; Section 5 discusses our evaluation; Section 6 analyzes our results; and Section 7 summarizes our work.

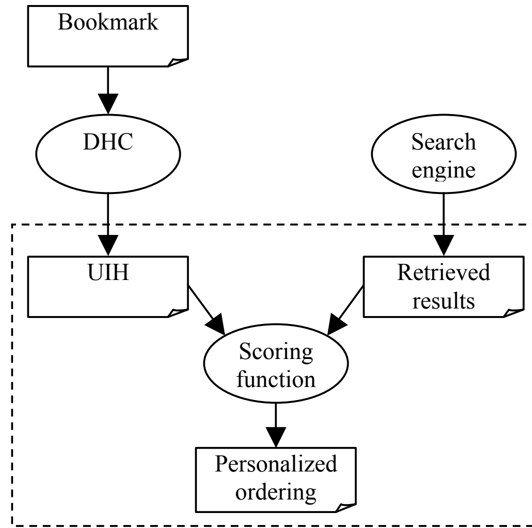

Figure 1. Diagram of Scoring

2. RELATED WORK

Page et al. [30] first proposed personalized web search by modifying the global PageRank algorithm with the input of bookmarks or homepages of a user. their work mainly focuses on global "importance" by taking advantage of the link structure of the web. Haveliwala [17] determined that PageRank could be computed for very large subgraphs of the web on machines with limited main memory. Brin et al. [3] suggested the idea of biasing the PageRank computation for the purpose of personalization, but it was never fully explored. Bharat and Mihaila [2] suggested an approach called *Hilltop*, that generates a query-specific authority score by detecting and indexing pages that appear to be good experts for certain keywords, based on their links. Hilltop is designed to improve results for *popular* queries; however, query terms for which experts were not found will not be handled by the Hilltop algorithm. Haveliwala [18] used personalized PageRank scores to enable "topic sensitive" web search. They concluded that the use of personalized PageRank scores can improve web search, but the number of hub vectors (e.g., number of interesting web pages used in a bookmark) used was limited to 16 due to the computational requirements. Jeh and Widom [19] scaled the number of hub pages beyond 16 for finer-grained personalization. Our method does not use the structure of hyperlinks.

Liu et al. [25] also tried mapping user queries to sets of categories. This set of categories served as a context to disambiguate the words in the user's query, which is similar to Vivisimo [36]. They studied how to supply, for each user, a small set of categories as a context for each query submitted by the user, based on his or her search history. Our approach does not personalize the set of categories, but personalizes results returned from a search engine.

Another approach to web personalization is to predict forward references based on partial knowledge about the history of the session. Zukerman et al. [40] and Cadez et al. [4] use a Markov model to learn and represent significant dependencies among page references. Shahabi and Banaei-Kashani [33] proposed a web-usage-mining framework using navigation pattern information. They introduced a feature-matrices (FM) model to discover and interpret users' access patterns. This approach is different from ours since we use the contents of web pages, and not navigation patterns.

PowerBookmarks [24] is a web information organization, sharing, and management tool, that monitors and utilizes users' access patterns to provide useful personalized services. PowerBookmarks provides automated URL bookmarking, document refreshing, bookmark expiration, and subscription services for new or updated documents. BookmarkOrganizer [26] is an automated system that maintains a hierarchical organization of a user's bookmarks using the classical HAC algorithm [37], but by applying "slicing" technique (slice the tree at regular intervals and collapse into one single level all levels between two slices). Both BookmarkOrganizer and PowerBookmarks reduce the effort required to maintain the bookmark, but they are insensitive to the context browsed by users and do not have reordering functions.

3. PERSONALIZED RESULTS

Personalization of web search involves adjusting search results for each user based on his or her unique interests. Our approach orders the pages returned by a search engine depending on a user's interests. Instead of creating our own web search engine, we retrieved results from Google [14]. Since the purpose of this paper is to achieve a personalized ordering of search engine

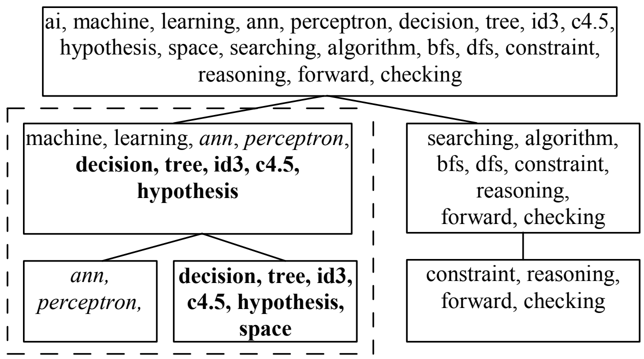

Figure 2. Sample user interest hierarchy

results, we can score a page based on the user profile and the results returned by a search engine as shown in the dashed box in Figure 1.

To build the user profile, called User Interest Hierarchy (UIH), we use the web pages in his/her bookmarks [24, 26] and the Divisive Hierarchy Clustering (DHC) algorithm [23]. A UIH organizes a user's interests from general to specific. Near the root of a UIH, general interests are represented by larger clusters of terms while towards the leaves, more specific interests are represented by smaller clusters of terms. The root node contains all distinct terms in the bookmarked web page. The leaf nodes contain more specifically interesting terms. The relations between terms are calculated based on the co-occurrence in the same web page.

An example of a UIH is shown in Figure 2. Each node (cluster) contains a set of words. The root node contains all words that exist in a set of web pages. Each node can represent a conceptual relationship if those terms occur together at the same web page frequently, for example 'perceptron' and 'ann' (in italics) can be categorized as belonging to neural network algorithms, whereas 'id3' and 'c4.5' (in bold) in another node cannot. Words in these two nodes are mutually related to some other words such as 'machine' and 'learning'. This set of mutual words, 'machine' and 'learning', performs the role of connecting italicized and bold words in sibling nodes and forms the parent node. We illustrate this notion in the dashed box.

This paper focuses on devising a scoring method that receives two inputs (UIH and retrieved results) and one output (personalized ranking).

4. APPROACH

In order to provide personalized, reordered search results to a user, we need to score each page depending on personal interests. Therefore, the goal is to assign higher scores to web pages that a user finds more interesting. This section explains how to score a retrieved web page using a user's UIH. First, we explain the basic characteristics for each matching term. Second, based on the characteristics, we propose functions to score a term. These functions determine how interesting a term is to a user. Third, based on the score and the number of the matching terms, we calculate an overall score for the page. Last, since the search engine provides a score/ranking for a web page, we incorporate this ranking into our final score of the web page.

4.1 Four Characteristics of a Term

Given a web page and a UIH, we identify matching terms (words/phrases) that reside both in the web page and in the UIH. The number of matching terms is defined m, which is less than the number of total distinct terms in the web page, n, and the number of total distinct terms in the UIH, l.

Each matching term, t_i, is analyzed according to four characteristics: the level of a node where a term belongs to (D_{t_i}), the length of a term such as how many words are in the term (L_{t_i}), the frequency of a term (F_{t_i}), and the emphasis of a term (E_{t_i}). D and L can be calculated while building a UIH from the web pages in a user's bookmark. Different web page has different values for F and E characteristics. We estimate the probability of these four characteristics and based on these probabilities, we approximate the significance of each matching term.

4.1.1 Level/depth of a UIH Node

A UIH represents general interests in large clusters of terms near the root of the UIH, while more specific interests are represented by smaller clusters of terms near the leaves. The root node contains all distinct terms and the leaf nodes contain small groups of terms that represent more specific interests. Therefore, terms in more specific interests are harder to match, and the level (depth) where the term matches indicates significance. For example, a document that contains terms in leaf nodes will be more related to the user's interests than a document that contains the terms in a root node only. If a term in a node also appears in several of its ancestors, we use the level (depth) closest to the leaves.

There is research that indicates user-defined query scores can be used effectively [32,16,7]. From the acquisition point of view, it is not clear how many levels of importance users can specify if we ask a user directly. In I^3R [8], they used only two levels: important or default. Harper [16] used 5 levels of importance, and Croft and Das [7] used 4 levels. We calculate the scores of terms using the level (depth) of a node in the UIH instead of explicitly asking the user.

The significance of a term match can be measured by estimating the probability, $P(D_{t_i})$, of matching term t_i at depth (level) D_{t_i} in the UIH. $P(D_{t_i})$ is the probability of a level in a UIH. A term that matches more specific interests (deeper in the UIH) has a lower $P(D_{t_i})$ of occurring. Lower probability indicates the matching term, t_i, is more significant. The probability is estimated by:

$$P(D_{t_i}) = \frac{\text{number of distinct terms at depth } D_{t_i} \text{ in the UIH}}{l}$$

4.1.2 Length of a Term

Longer terms (phrases) are more specific than shorter ones. If a web page contains a long search term typed in by a user, the web page is more likely what the user was looking for.

In general, there are fewer long terms than short terms. To measure the significance of a term match, the probability, $P(L_{t_i})$, of matching term t_i of length L_{t_i} in the UIH is calculated. L_{t_i} is defined as $MIN(10, \text{the length of a term})$. We group the longer (greater than 10) phrases into one bin because they are rare. Longer terms has a smaller probability, $P(L_{t_i})$, of occurring, which indicates a more significant match. The probability is estimated by:

$$P(L_{t_i}) = \frac{\text{number of distinct terms of length } L_{t_i} \text{ in the UIH}}{l}$$

4.1.3 *Frequency of a Term*

More frequent terms are more significant/important than less frequent terms. Frequent terms are often used for document clustering or information retrieval [35]. A document that contains a search term many times will be more related to a user's interest than a document that has the term only once.

We estimate the probability, $P(F_{t_i})$, of a matching term t_i at frequency F_{t_i} in a web page to measure the significance of the term. However, in general, frequent terms have a lower probability of occurring. For example, in a web page most of the terms (without the terms in a stop list [12]) will occur once, some terms happen twice, and fewer terms repeat three times or more. Lower probabilities, $P(F_{t_i})$, of a term t_i indicates the significance of a term. The probability is estimated by:

$$P(F_{t_i}) = \frac{\text{number of distinct terms with frequency } F_{t_i} \text{ in a web page}}{n}$$

4.1.4 *Emphasis of a Term*

Some terms have different formatting (HTML tags) such as title, bold, or italic. These specially-formatted terms have more emphasis in the page than those that are formatted normally. A document that emphasize a search term as a bold format will be more related to the search term than a document that has the term in a normal format without emphasis. If a term is emphasized by the use of two or more types of special formatting we assign a priority in the order of title, bold, and italic.

The significance for each type of format is estimated based on the probability, $P(E_{t_i})$, of matching term t_i with the format type E_{t_i} in a web page. Those format types are more significant/important if the format type has lower probability of occurring in a web page. Lower probability $P(E_{t_i})$ of a matching term, t_i, indicates the term is more significant. The probability is estimated by:

$$P(E_{t_i}) = \frac{\text{number of distinct terms with emphasis } E_{t_i} \text{ in a web page}}{n}$$

4.2 Scoring a Term

4.2.1 *Uniform Scoring*

$P(D_{t_i}, L_{t_i}, F_{t_i}, E_{t_i})$ is the joint probability of all four characteristics occurring in term t_i -- D_{t_i} is the depth of a node where a term belongs to, L_{t_i} is the length of a term, F_{t_i} is the frequency of a term, and E_{t_i} is the emphasis of a term. Assuming independence among the four characteristics, we estimate:

$$P(D_{t_i}, L_{t_i}, F_{t_i}, E_{t_i}) = P(D_{t_i}) \times P(L_{t_i}) \times P(F_{t_i}) \times P(E_{t_i})$$

The corresponding log likelihood is:

$$\log P(D_{t_i}, L_{t_i}, F_{t_i}, E_{t_i}) = \log P(D_{t_i}) + \log P(L_{t_i}) + \log P(F_{t_i}) + \log P(E_{t_i}) \qquad \text{Eq. 1}$$

Smaller log likelihood means the term match is more significant. In information theory [28], $-\log_2 P(e)$ is the number of bits needed to encode event e, hence using $-\log_2$, instead of log, in Eq. 1 yields the total number of bits needed to encode the four characteristics. The uniform term scoring (US) function for a personalized term score is formulated as:

$$S_{t_i} = -\log_2 P(D_{t_i}) - \log_2 P(L_{t_i}) - \log_2 P(F_{t_i}) - \log_2 P(E_{t_i}) \qquad \text{Eq. 2}$$

which we use as a score for term t_i. Larger S_{t_i} means the term match is more significant.

4.2.2 *Weighted Scoring*

The uniform term scoring function uses uniform weights for each characteristic. It is possible that some characteristics are more important than the others. For instance, the depth of a node (*D*) may be more significant than frequency (*F*). Therefore, we attempted to differentiate the weights for each characteristic. *F* and *E* characteristics represent the relevance of a web page. Longer terms (greater *L*) *r*epresent a user's interest more specifically; however, longer terms do not mean that a user is more interested in that term. Therefore, those *L*, *F*, and *E* characteristics do not fully reflect a user's interests. It is more reasonable to emphasize *D* characteristic more than other characteristics, because *D* (depth) represents the strength of a user's interests.

A simple heuristic is used in this paper that assumes the depth of a node is at least two times more important than other characteristics. Based on this heuristic, the weights w_1=0.4, w_2=0.2, w_3=0.2, and w_4=0.2 are assigned. The weighted term scoring (WS) function for a personalized term score is formulated as:

$$S_{t_i} = -w_1 \log_2 P(D_{t_i}) - w_2 \log_2 P(L_{t_i}) - w_3 \log_2 P(F_{t_i}) - w_4 \log_2 P(E_{t_i}) \qquad \text{Eq. 3}$$

4.3 Scoring a Page

The *personal* page score is based on the number of interesting terms and how interesting the terms are in a web page. If there are many terms in a web page that are interesting to a user, it will be more interesting to the user than a web page that has fewer interesting terms. If there are terms in a pages that are more interesting to a user, the web page will be more interesting to the user than a web page that has less interesting terms.

The personalized page scoring function for a web page S_{p_j} adds all the scores of the terms in the web page and can be formulated as:

$$S_{p_j} = \sum_{i=1}^{m} S_{t_i} \qquad \text{Eq. 4}$$

where *m* is the total number of matching terms in a web page and S_{t_i} is the score for each distinct term. The time complexity of scoring a page is $O(n)$, where *n* is the number of "distinct" terms in a web page. *D* and *L* characteristics can be calculated during the preprocessing stage of building a UIH. *F* and *L* characteristics can be calculated while extracting distinct terms from a web page.

4.4 Incorporating Public Page Score

Personal page scoring is not sufficient for some search engines. The success of using *public* scoring in popular search engines, such as Google's PageRank, indicates the importance of using a public page-popularity measure to determine what page a user is interested in. Many existing methods determine the public popularity of a page by determining the number pages that link to it [18,19]. Many collaborative filtering approaches also use the popularity of a web page for recommendation [11,20]. Section 4.3 described our personal web page scoring function. We wish to incorporate the *public* scoring into our page scoring function so both the popularity of a page and individual interests are taken into account. We use the rank order returned by Google as our *public* score. $GOOGLE_{p_j}$ is the score of a web page p_j based on the page rank returned by Google for a search term. Google's *public* page scoring function has been found in a recent study [9] to be very effective at returning pages that users find interesting. The use of Google's page rank as a *public* page score makes our experimental comparison with Google clearer, because any improvement in the ordering is due to the contribution of our *personal* page score. For a given web page, p_j, the *personal* and *public* page score (*PPS*) equation can be written as:

$$PPS_{p_j} = c \times R\,(S_{p_j}) + (1\text{-}c) \times R\,(GOOGLE_{p_j}) \qquad \text{Eq. 5}$$

where function $R(GOOGLE_{p_j})$ return the rank of a web page, p_j, with the *public* page score of $GOOGLE_{p_j}$, and $R(S_{p_j})$ is the rank of a web page, p_j, with the *personal* page score, S_{p_j}. If the function R returns the rank in an ascending order, more interesting web pages will have lower *PPS* values. Therefore, the function R reverses the rank. The *personal* page score and the *public* page score are weighted by the value of the constant c. In this paper, both functions are weighed equally: $c = 0.5$.

5. EXPERIMENTS

In our experiments data were collected from 11 different users. Of the 11 human subjects, 4 were undergraduate students and 7 were graduate students. In terms of major, 7 were Computer Sciences, 2 were Aeronautical Sciences, 1 was Chemical Engineering, and 1 was Marine Biology. We asked each volunteer to submit 2 search terms that can contain any Boolean operators. Some examples of the search terms used are

{`review forum +"scratch remover", cpu benchmark, aeronautical, Free cross-stitch scenic patterns, neural networks tutorial, DMC(digital media center), artificial intelligence`, etc.}

Then, we used Google to retrieve 100 related web pages for each search term. Those collected web pages were classified/labeled by user based on two categories: *interest* and *potential interest*. The data set for *interest* has more authority because it indicates direct relevance to the current search query. The data set for *potential interest* reflects the user's general personal interests, which might not be directly relevant at the time of query. The areas of a user's *potential* interests often go beyond the boundary of a search term's specific meaning. Sometimes users find interesting web pages while searching for different subjects. These unexpected results help the user as well. Therefore, it is also a contribution if a method shows higher precision in finding *potentially* interesting web pages.

In order to build UIHs, we also requested each volunteer to submit the web pages in their bookmarks. If there were fewer than 50 web pages in their bookmark, we asked them to collect more pages up to around 50. The minimum number of web pages was 38 and the maximum number was 72. Web pages from both bookmarks and Google were parsed to retrieve only texts. The terms (words and phrases) in the web pages are stemmed and filtered through the stop list [12]. A phrase-finding algorithm [22] was used to collect variable-length phrases. Words in selection boxes/menus were also removed because they did not appear on the screen until a user clicks on them. Unimportant contexts such as comments and style were also removed. Web pages that contain non-text (e.g., ".pdf" files, image files, etc.) were excluded because we are handling only text. To remove any negative bias to Google, broken links that were still ranked high erroneously by Google were excluded from the test, since those web pages will be scored "Poor" by the user for sure. The data used in this study is accessible at http://cs.fit.edu/~hkim/dissertation/dissertation.htm.
Microsoft .NET language was used, and the program ran on an Intel Pentium 4 CPU.

We attempted to remove any negative bias to Google. Those web pages that contain non-text (e.g., ".pdf" files, image files, etc.) were excluded because we are handling only texts. Furthermore, the broken links that were still ranked high erroneously by Google were excluded from the test, since those web pages will be scored "Poor" by user for sure.

We categorized the *interest* as "Good", "Fair", and "Poor"; the *potential interest* is categorized as "Yes" and "No". A web page was scored as "Good", "Fair", and "Poor" depending on each individual's subjective opinion based on the definition of *interest*. It was also marked as "Yes" or "No" based on the user's *potential interest*. We evaluated a ranking method based on how many interesting (categorized as "Good") or *potentially* interesting web pages (categorized as "Yes") the method collected within a certain number of top links [2] (called "Top link analysis"). It is realistic in a sense many information retrieval systems are interested in the top 10 or 20 groups. Precision/recall graph [35] is used for evaluation as well (called "precision/recall analysis"). It is one of the most common evaluation methods in information retrieval. However, traditional precision/recall graphs are very sensitive to the initial rank positions and evaluate entire rankings [7]. The formula for precision and recall were:

```
Precision = Number of "Good" or "Yes" pages
            retrieved in the set / Size of the
            set
Recall    = Number of "Good" or "Yes" pages
            retrieved in the set / Number of
            "Good" or "Yes" pages in the set
```

where the "set" is the group of top ranked web pages. In this paper we study five groups: Top 1, 5, 10, 15, and 20.

6. ANALYSIS

We compare four ranking methods: Google, Random, US, and WS. Google is the ranking provided by Google. Random arbitrarily ranks the web pages. US and WS are the two proposed methods based on a personal UIH learned from a user's bookmarks. For Random, US, and WS, the top 100 pages retrieved by Google are re-ranked based on the method. Each method is analyzed with two data sets: a set of web pages chosen

Table 1. Precision in Top 1, 5, 10, 15 and 20 for interesting web pages

	Top 1	Top 5	Top 10	Top 15	Top 20
Google	<u>.36</u>	<u>.34</u>	.277	.285	.270
Random	.14	.25	.205	.206	.209
US	.32	.31	<u>**.323 (17%)**</u>	**.315 (11%)**	**.305 (13%)**
WS	<u>.36</u>	<u>.34</u>	**.314 (13%)**	<u>**.327 (15%)**</u>	<u>**.309 (14%)**</u>

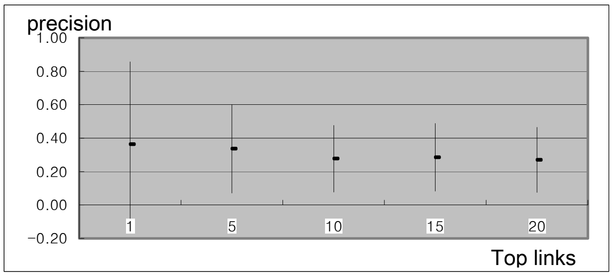

Figure 3. Average and SD of precision with Google

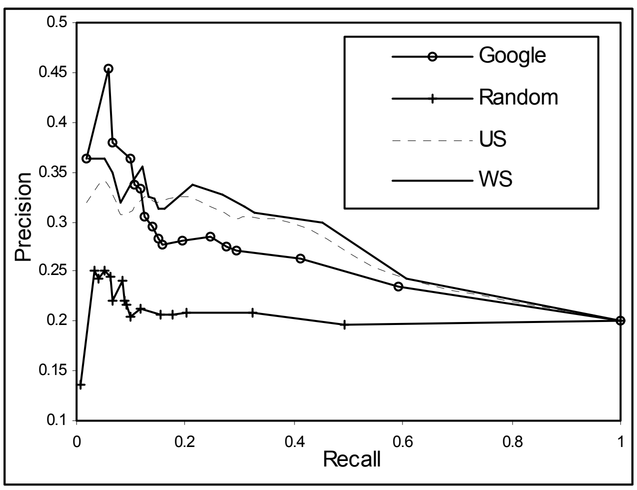

Figure 4. Precision/recall graph for interesting web pages

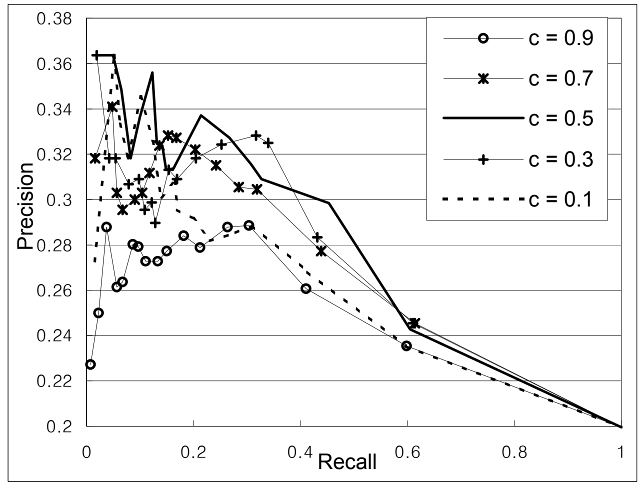

Figure 5. Precision/recall with personal score weight *c*

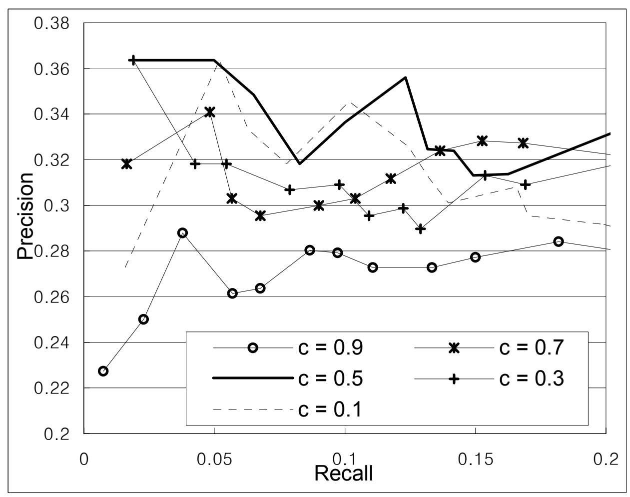

Figure 6. Up to 20% recall of Figure 5

as interesting and another chosen as potentially interesting by the users. Top link analysis, precision/recall analysis, the sensitivity of personal score weight *c* (Section 4.4) are discussed.

6.1 Interesting Web Page

6.1.1 Top Link Analysis

Web search engine users are usually interested in the links ranked within top 20 [6]. We compare each method only with Top 1, 5, 10, 15, and 20 links on the *interesting* web page data set and present the results in Table 1. The first column is the methods; the next five columns present the precision values of each method with respect to the five Top links. The values in each cell are the average of 22 search terms' precision values. High precision value indicates high accuracy/performance. Precision values higher than Google's are formatted as bold and the percentage of improvement is within parentheses. The highest precision value in each column is underscored.

The results show that our WS method was more accurate than Google in three Top links (Top 10, 15, and 20) and the percentages of improvements are at least 13%, while WS ties with Google for Top 1 and Top 5. In terms of the highest precision, WS showed highest performance in four columns; Google showed in only two columns and the values are equal to WS. Compared to US, WS showed higher precision in four (Top 1, 5, 15 and 20) of the five columns. Random was the lowest as we expected, showing the lowest precisions in all five columns. These results indicate that WS achieves the highest overall precision.

We also wanted to know which search terms yielded higher precision with WS than with Google and analyzed the precision with respect to each individual search terms. Out of 22 search terms (11 users × 2 search terms), WS achieved higher precision for 12 search terms (55%), Google did for 8 search terms (36%), and they were even for 2 search terms (9%). Since the UIH is built from a user's bookmarks, we analyse the bookmarks to understand the search terms that did not perform well using WS. When we compare the bookmarks with the "good" retrieved web pages, we found that they are unrelated. For example, a volunteer used "woodworking tutorial" as a search term, but he never bookmarked web pages related to that term. This implies bookmarks are useful for building user profiles, but they are not sufficient. We will discuss enhancements in the conclusion.

6.1.2 Statistical Significance

In order to see if this improvement is statistically significant we conducted t-Test (paired two samples for means) between two groups of individual search terms with Google and WS for each Top link. There was no statistically significant difference between WS and Google for any Top link with 95% confidence (P=1and t=2.079 for Top 1; P=1and t=2.079 for Top 5; P=0.328 and t=2.079 for Top 10; P=0.204 and t=2.079 for Top 15; P=0.147 and t=2.079 for Top 20).

To understand why our improvements are not statistically significant, we analyze the variance in the precision values. In Figure 3 we plot the average and the standard deviation (SDs) of 22 search terms' precisions from Google with respect to the five Top links. The x-axis shows the Top links and y-axis represents the average and the SD of precision values. The dots in the middle of vertical bars are the averages and the bars themselves represent the SD values. Variance was large for Top 1 and decreases when more links were considered.

To understand the difficulty of improving Google's ranking, we calculate the number of multiples needed to achieve one SD from the average. Formally, the number of multiples is defined as (Avg.+SD)/Avg. The larger the number of multiples indicates more difficulty in beating Google with statistically significance. The number of multiples for Top 1 is 3.35, Top 5 is 2.79, Top 10 is 2.72, Top 15 is 2.71, and Top 20 is 2.72. In information retrieval doubling or tripling the precision for a large variance like Google's is rare. From our calculated number of multiples, we need to at least double or triple the precision to achieve statistically significant improvement over Google's ranking.

To further demonstrate the difficulty, we applied the same t-Test to precision values from Google and Random (P=0.134, t=2.079 for Top 1; P=0.179, t=2.079 for Top 5; P=0.062, t=2.079 for Top 10; P=0.035, t=2.079 for Top 15; P=0.024, t=2.079 for Top 20). We found that, though Google's improvement over random is statistically significant for Top 15 and 20, it is *not* statistically significant for Top 1, 5, and 10.

6.1.3 Precision/Recall Analysis

Precision/recall analysis visualizes the performance of each method in graphs as shown in Figure 4. The x-axis is recall and y-axis is precision. The line closer to the upper-right corner has higher performance. WS and US are closer to the upper-right corner than Google except with recall values lower than .15 (after Top 5). In general, WS outperforms US and Random.

6.1.4 Varying Personal Weight

The performance of WS may depend on how much we weigh the *personal* page score over the *public* page score. The parameter c in Section 4.4 represents the weight for the *personal* page score. For example, c=0.9 means the page is scored by 90% of *personal* page score and 10% of *public* page score. We experimented with c={0.9, 0.7, 0.5, 0.3, and 0.1} and measured the corresponding precision and recall. The results are plotted in Figure 5 and each line represents a different c value. The line closer to the upper right corner indicates higher performance. c=0.9 has lowest precision. c=0.1 achieved the second lowest precision except for recall values lower than 0.2.

Figure 6 enlarges the scale of recall between 0 through 0.2 in Figure 5. It is still not clear which one is higher than the others except the line with c=0.9; however, c=0.5 in general seems to show the highest performance. Therefore, we chose c=0.5 as the weight of *personal* page score.

6.2 Potentially Interesting Web Page

6.2.1 Top Link Analysis

We compare our four methods with Top 1, 5, 10, 15, and 20 links on the *potentially* interesting web page data set and present the results in Table 2. The values in each cell are the average of 22 search terms' precision values. The ways of reading this table and the table for *interesting* web pages are similar.

WS showed higher performances than Google in all five Top links. All five precisions achieved by WS are the highest values as well. The percentages of improvements are between 3% and 17%. Random showed the lowest in all five Top links. The reason for the improvement of WS is, we predict, because the UIH that was derived from a user's bookmarks supported the user's *potential* interest. It might be difficult for Google that used the *global*/*public* interest to predict individual user's broad *potential* interests.

We also counted what search terms yielded higher precision with WS than with Google. WS achieved higher performance for 12 search terms (55%), Google made for 8 search terms (36%), and they were even for 2 search terms (9%) out of 22 search terms. The reason for the low performance of some search terms might be because there is no relation between his/her bookmarks and the search terms.

6.2.2 Statistical Significance

The t-Test between the two groups of 22 individual search terms with WS and Google showed no statistically significant difference with 95% confidence for any Top link (P=0.665 and t=2.079 for Top 1; P=0.115 and t=2079 for Top 5; P=0.466 and t=2.079 for Top 10; P=0.580 and t=2.079 for Top 15; P=0.347 and t=2.079 for Top 20).

We analyze the variance in the precision values to understand why the improvements are not statistically significant. The graph in Figure 7 illustrates the average and the standard deviation (SD) of 22 search terms' precisions with Google to the five Top links. Variance was large for Top 1 and decreased as more links were considered.

Table 2. Precision in Top 1, 5, 10, 15 and 20 for potentially interesting web pages

	Top 1	Top 5	Top 10	Top 15	Top 20
Google	.59	.53	.514	.509	.475
Random	.36	.39	.350	.358	.364
US	.59	**.58 (9%)**	**.536 (4%)**	**.521 (2%)**	**.493 (4%)**
WS	**<u>.64 (8%)</u>**	**<u>.62 (17%)</u>**	**<u>.541 (5%)</u>**	**<u>.524 (3%)</u>**	**<u>.498 (5%)</u>**

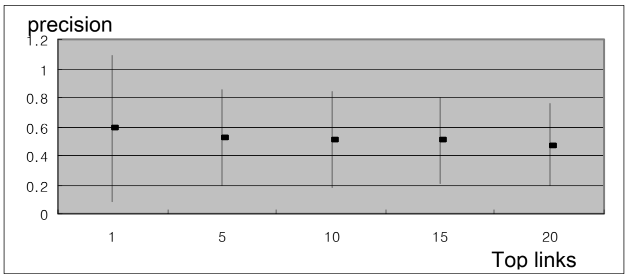

Figure 7. Average and SD of precision with Google

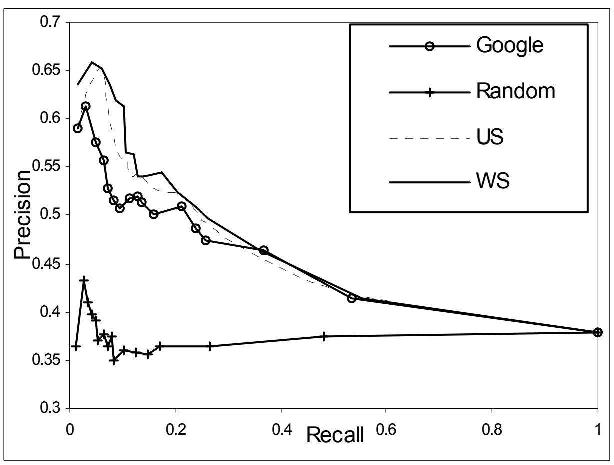

Figure 8. Precision/recall graph for potentially interesting web pages

In order to estimate how difficult it is to achieve statistically significant improvements over Google, we calculate the number of multiples, which is (Avg.+SD)/Avg. The number of multiples for Top 1 is 2.77, for Top 5 is 2.53, for Top 10 is 2.59, for Top 15 is 2.58, for Top 20 is 2.60. From our calculated number of multiples, we need to at least double the precision for achieving statistically significant improvement on *potentially* interesting web pages.

6.2.3 Precision/Recall Analysis

The results from precision/recall graph for *potentially* interesting web pages in Figure 8 and the Top link analysis in Table 2 are similar. WS was closer to the upper-right corner than Google, US, and Random over all. WS outperformed other methods on *potentially* interesting web pages data set.

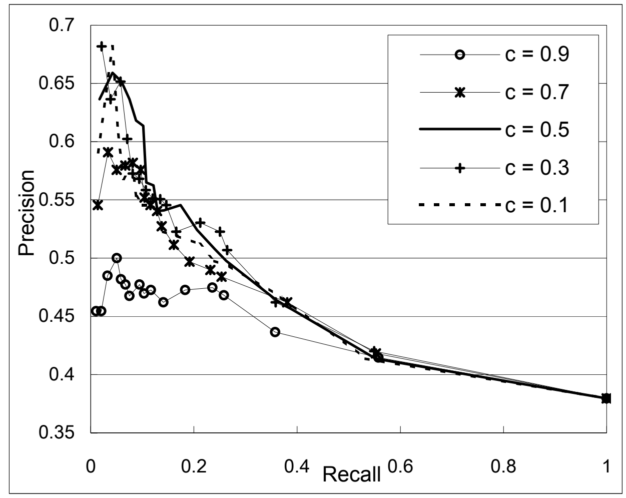

Figure 9. Precision/recall with personal score weight *c*

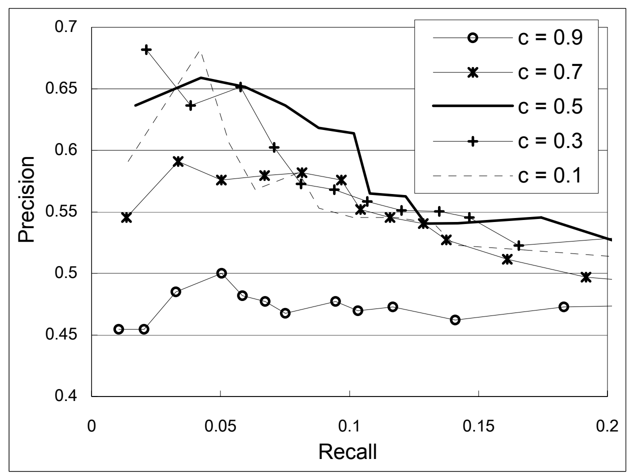

Figure 10. Up to 20% recall of Figure 9

6.2.4 Varying Personal Weight

In order to see the improvement of WC's performance with different *personal* weights, we varied the parameter for the weight: c = {0.9, 0.7, 0.5, 0.3, and 0.1}. The results are plotted in Figure 9. c=0.9 draw the lowest line; c=0.7 looks the second lowest line.

Figure 10 enlarges the scale of recall in Figure 9. It looks clear that c=0.5 in general seems to show the highest performance. Therefore, we chose c=0.5 for the weight of *personal* page score. The weights for personal page score with both data sets are the same.

7. CONCLUSION

The purpose of this research is to devise a new method of ranking web search results to serve each individual user's interests. A user profile called UIH is learned from his/her bookmarks. For scoring a term in a web page that matches a term in the UIH, we identified four characteristics: the depth of tree node in the UIH that contains the term, the length of the term, the frequency of the term in the web page, and the html formatting used for emphasis. Our approach uses the terms filtered though stop list in web pages [12]. This approach removes the process of selecting important/significant terms unlike other information retrieval techniques [31]. Therefore, we can handle smaller data set and reduce the danger of eliminating new important terms. We evaluated methods based on how many interesting web pages or potentially interesting web pages each algorithm found within certain number of top links [2]. Traditional precision/recall graphs [35] were also used for evaluation. We counted which search term showed higher performances with WS than with Google as well.

We compared four ranking methods: Google, Random, US, and WS. Google is the most popular search engine and posts the best ordering results currently. Random method was chosen to see the improved performance of Google and our new methods. We used two data sets: interesting web pages that are relevant to the user search term and potentially interesting web pages that could be relevant in the future. On interesting web pages, the Top link analysis indicated WS achieved at least 13% higher precision than Google for Top 10, 15 and 20 links on average. WS outperformed US and Random in general also. The precision/recall analysis showed that WS outperformed Google except with recall values lower than .15. Out of 22 search terms, WS achieved higher precision than Google for 12 search terms (55%). On potentially interesting web pages, WS achieved the highest performance in all five Top links with improvement over Google between 3% and 17%. It also outperformed the other methods in the precision/recall graph. The analysis of individual search terms yielded the same results as on interesting web pages. A weight of 0.5 for the personal ranking seemed to show the highest performance on both data sets. Therefore, these results conclude that WS can provide more accurate ranking than Google on average. The improvement of WS was not statistically significant because the precision values of Google had large variance. The reason for the low performance of some search terms might be because there is no relation between his/her bookmarks and the search terms. We may be able to relieve this problem by incorporating interesting web pages based on implicit interest indicators such as mouse movements [21] in addition to bookmarking.

During the experiment, we observed that users do not tend to measure index pages as "Good". It is because index pages usually contain long lists of hyperlinks with little description for a user to find interesting. To identify index pages automatically, we count the number of "outside words" (the text outside anchor tags), which usually provide the subject content. However, our approach of penalizing the index pages did not make much improvement in our initial experiments. We will examine this approach further in the future.

Measuring the precision with clustered search results like the results from Vivisimo [36] may show different performance from Google's. In a clustered search engine, a link that does not belong to the top 10 in whole can belong to the top 10 in some sub clusters. The clustered search results provide users easier access to the *interesting* links after Top 10 or 20. Since WS showed higher performance for those links than Google as shown in Section 6.1.3, we assume that our method may get higher performance with clustered search engines. We may be able to make a search engine more interactive using a UIH. For example when a user's query resides in an intermediate node in his/her UIH, we can ask a user to choose more specific interests providing the terms in the child nodes, or in another sub-trees in the UIH.

8. ACKNOWLEDGMENTS

We appreciate all volunteers who participated in our experiment: Akiki, Michel, Timmy, Matthew Scripter, Ayanna, Da-hee Jung, Jae-gon Park, Ji-hoon Choi, Jun-on, Chris Tanner, and Grant Beems.

9. REFERENCES

1 Anderson, C.R. *A Machine Learning Approach to Web Personalization*, Ph.D. thesis. University of Washington, Department of Computer Science and Engineering, 2002.

2 Bharat, K. and Mihaila, G. A. When experts agree: using non-affiliated experts to rank popular topics. In *Proceedings of the 10th Intl. World Wide Web Conference*, 2001.

3 Brin, S., Motwani, R., Page, L., and Winograd, T. What can you do with a web in your pocket. In *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, 1998.

4 Cadez, I., Heckerman, D., Meek, C., Smyth, P., and White, S. *Visualization of Navigation Patterns on Web Site Using Model Based Clustering.* Technical Report MSR-TR-00-18, Microsoft Research, Microsoft Corporation, Redmond, WA, 2000.

5 Chan, P.K. A non-invasive learning approach to building web user profiles, In *KDD-99 Workshop on Web Usage Analysis and User Profiling*, 1999, 7-12.

6 Chen, L. and Sycara, K. WebMate: A personal agent for browsing and searching. In *Proc. of the 2nd Intl. conf. on Autonomous Agents*, 1998,132-139.

7 Croft, W. B. and Das, R. Experiments with query acquisition and use in document retrieval systems. In *Proc. of 13th ACM SIGIR*, 1989.

8 Croft, W. B. and Thompson, R. T. I3R: A new approach to the design of document retrieval systems. *Journal of the Americal Society for Information Science*, 38 (1987), 389-404.

9 Delaney, K. J. Study questions whether google really is better. *Wall Street Journal*. (Eastern edition). New York, May 25, 2004, B.1. http://proquest.umi.com/pqdweb?RQT=309&VInst=PROD&VName=PQD&VType=PQD&sid=5&index=45&SrchMode=1&Fmt=3&did=000000641646571&clientId=15106

10 Deshpande, M. and Karypis, G. Selective markov models for predicting web-page accesses. *First SIAM International Conference on Data Mining*, 2001.

11 Eirinaki, M., Lampos, C., Paulakis, S., and Vazirgiannis M. Web personalization integrating content semantics and navigational patterns. *Workshop on Web Information and Data Management*, 2004, 72 – 79.

12 Frakes, W.B., and Baeza-Yates, R. Information Retrieval: Data Structures and Algorithms, Prentice-Hall, 1992.

13 Fu, X., Budzik, J., and Hammond, K.J. Mining navigation history for recommendation, In *Proc. 2000 Conference on Intelligent User Interfaces*, 2000.

14 Google co. 2004. http://www.google.com/

15 Grossman, D., Frieder, O., Holmes, D., and Roberts, D. Integrating structured data and text: A relational approach. *Journal of the American Society for Information Science*, 48, 2 (February 1997).

16 Harper, D. J. *Relevance Feedback in Document Retrieval Systems: An Evaluation of Probabilistic Strategies*. Ph.D. Thesis, Computer Laboratory, University of Cambridge, 1980.

17 Haveliwala, T. H. *Efficient computation of PageRank*. Technical Report, Stanford University Database Group, 1999. http://dbpubs.stanford.edu/pub/1999-31

18 Haveliwala, T. H. Topic-sensitive PageRank. In *Proc. of the 11th Intl. World Wide Web Conference*, Honolulu, Hawaii, May 2002.

19 Jeh, G. and Widom, J. Scaling personalized web search. In *Proc. of the 12th Intl. Conference on World Wide Web*, Budapest, Hungary, pp. 20-24, May 2003.

20 Kim, D., Atluri, V., Bieber, M., Adam, N., and Yesha, Y. A clickstream-based collaborative filtering personalization model: towards a better performance. *Workshop on Web Information and Data Management*, 2004.

21 Kim, H. and Chan, P. K. *Implicit indicator for interesting web pages*. Technical Report CS-2005-05, Florida Institute of Technology, 2005.

22 Kim, H. and Chan, P. K. Identifying variable-length meaningful phrases with correlation functions, *IEEE International Conference on Tools with Artificial Intelligence*, IEEE press, 2004, 30-38.

23 Kim, H. and Chan, P. K. Learning implicit user interest hierarchy for context in personalization. *International Conference on Intelligent User Interfaces*, 2003, 101-108.

24 Li, W. S., Vu, Q., Agrawal, D., Hara, Y., and Takano, H. PowerBookmarks: A System for personalizable web information organization, sharing, and management. In *Proc. of the 8th Intl. World Wide Web Conference*, Toronto, Canada, 1999.

25 Liu, F., Yu, C., and Meng, W. Personalized web search by mapping user queries to categories. *CIKM'02*, ACM Press, Virginia, USA, 2002.

26 Maarek, Y. S. and Ben-Shaul, I. Z. Automatically organizing bookmarks per contents. In *Proceedings of the Fifth International World Wide Web Conference*, 1996.

27 Maarek, Y.S. and Ben-Shaul, I.Z. Automatically Organizing Bookmarks Per Contents, *Proc. 5th International World Wide Web Conference*, 1996.

28 Mitchell, T. M. *Machine Learning*. New York: McGraw Hill, 1997.

29 Mobasher, B. Cooley, R. and Srivastave, J. Creating Adaptive Web Sites through Usage-Based Clustering of URLs, *Proc. 1999 IEEE Knowledge and Data Engineering Exchange Workshop*, 1999, 19-25.

30 Page, L., Brin, S., Motwani, R., and Winograd, T. *The PageRank citation ranking: Bringing order to the web*. Technical Report, Stanford University Database Group, 1998. http://citeseer.nj.nec.com/368196.html

31 Pazzani, M., and Billsus, D. Learning and revising user profiles: The identification of interesting web sites. *Machine Learning*, 27, 3 (1997), 313-331.

32 Salton, G. and Waldstein, R. G. Term relevance weights in on-line information retrieval. *Information Processing and Management*, 14 (1978), 29-35.

33 Shahabi, C. and Banaei-Kashani, F. Efficient and anonymous web-usage mining for web personalization. *INFORMS Journal on Computing-Special Issue on Data Mining*, 15, 2 (Spring 2003).

34 Stefani, A. and Strapparava, C. Exploiting nlp techniques to build user model for web sites: The use of worldnet in SiteIF project, *Proc. 2nd Workshop on Adaptive Systems and User Modeling on the WWW*, 1999.

35 van Rijsbergen, C. J. *Information Retrieval*. Butterworths, London, 1979, 68–176.

36 Vivisimo co. 2004 http://www.vivisimo.com

37 Voorhees, E. M. Implementing agglomerative hierarchic clustering algorithms for use in document retrieval. *Information Processing & Management*, 22, 6 (1986), 465-476.

38 Wexelblat, A. and Maes, P. Footprints: History-rich web browsing, *Proc. Conference on Computer-Assisted Information Retrieval* (*RIAO*), 1997, 75-84.

39 Yan, T.W. Jacobsen, M. Garcia-Molina, H. and Dayal, U. From user access patterns to dynamic hypertext linking, *Proc. 5th International World Wide Web Conference*, 1996.

40 Zukerman, I., Albrecht, D. W., and Nicholson, A. E. Predicting users' requests on the WWW. *Proc. of the 7th Intel. Conference on User Modeling* (*UM*), Banff, Canada, 1999, 275-284.