Personalized Search Based on User Search Histories

Mirco Speretta
Electrical Engineering and Computer Science
University Of Kansas
Lawrence, KS 66045
mirco@ku.edu

Susan Gauch
Electrical Engineering and Computer Science
University Of Kansas
Lawrence, KS 66045
+1 (785) 864-7755
sgauch@ku.edu

Abstract

User profiles, descriptions of user interests, can be used by search engines to provide personalized search results. Many approaches to creating user profiles collect user information through proxy servers (to capture browsing histories) or desktop bots (to capture activities on a personal computer). Both these techniques require participation of the user to install the proxy server or the bot. In this study, we explore the use of a less-invasive means of gathering user information for personalized search. In particular, we build user profiles based on activity at the search site itself and study the use of these profiles to provide personalized search results. By implementing a wrapper around the Google [10] search engine, we were able to collect information about individual user search activities. In particular, we collected the queries for which at least one search result was examined, and the snippets (titles and summaries) for each examined result. User profiles were created by classifying the collected information (queries or snippets) into concepts in a reference concept hierarchy. These profiles were then used to re-rank the search results and the rank-order of the user-examined results before and after re-ranking were compared. Our study found that user profiles based on queries were as effective as those based on snippets. We also found that our personalized re-ranking resulted in a 34% improvement in the rank-order of the user-selected results.

1. Introduction

Companies that provide marketing data report that search engines are utilized more and more as referrals to web sites, rather than direct navigation via web links [20]. As search engines perform a larger role in commercial applications, the desire to increase their effectiveness grows. However, search engines order their results based on the small amount of information available in the user's queries and by web site popularity, rather than individual user interests. Thus, all users see the same results for the same query, even if they have wildly different interests and backgrounds. To address this issue, interest in personalized search had grown in the last several years, and user profile construction is an important component of any personalization system. Explicit customization has been widely used to personalize the look and content of many web sites, but we concentrate on personalized search approaches that focus on implicitly building and exploiting user profiles.

Another issue facing search engines is that natural language queries are inherently ambiguous. For example, consider a user issuing the query "canon book". Due to the ambiguity of the query terms, we will obtain results that are either related to religion or photography. According to an analysis of their log file data conducted by OneStat.com [17] over a 2 month period of time, the most common query length submitted to a search engine (32.6 %) was only two words long and 77.2% of all queries were three words long or less. These short queries are often ambiguous, providing little information to a search engine on which to base its selection of the most relevant Web pages among millions. A user profile that represents the interests of a specific user can be used to supplement information about the search that, currently, is represented only by the query itself. This information could be used to narrow down the number of topics considered when retrieving the results, increasing the likelihood of including the most interesting results from the user's perspective. For the user in our example, if we knew that she had a strong interest in photography but little or none in religion, the photography-related results could be preferentially presented to the user.

Many approaches create user profiles by capturing browsing histories through proxy servers or desktop activities through the installation of bots on a personal computer. These require the participation of the user in order to install the proxy server or the bot. In this study, we explore the use of a less-invasive means of gathering user information for personalized search. Our goal is to show that user profiles can be implicitly created out of short phrases such as queries and snippets collected by the search engine itself. We demonstrate that profiles created from this information can be used to identify, and promote, relevant results for individual users.

Proceedings of the 2005 IEEE/WIC/ACM International Conference on Web Intelligence (WI'05)
0-7695-2415-X/05 $20.00 © 2005 **IEEE**

2. Background

Personalization is the process of presenting the right information to the right user at the right moment. Systems can learn about user's interests collecting personal information, analyzing the information, and storing the results in a user profile. Information can be captured from users in two ways: explicitly, for example asking for feedback such as preferences or ratings; and implicitly, for example observing user behaviors such as the time spent reading an online document. Explicit construction of user profiles has several drawbacks. The user provides inconsistent or incorrect information, the profile created is static whereas the user's interests may change over time, and the construction of the profile places a burden on the user that she may not wish to accept. Thus, many research efforts are underway to implicitly create accurate user profiles [1][6][7][18]. To achieve effective personalization, profiles should distinguish between long-term and short-term interests and include a model of the user's context, i.e., the task in which the user is currently engaged and the environment in which they are situated [15].

User browsing histories are the most frequently used source of information about user interests. Trajkova and Gauch [21] use this source to create user profiles represented as weighted concept hierarchies. User profiles are created by classifying the collected Web pages with respect to a reference ontology. Kim and Chan [13] also build user profiles from browsing histories, however they use clustering to create a user interest hierarchy. The collected Web pages are then assigned to the appropriate cluster. The fact that a user has visited a page is an indication of user interest in that page's content. Extending this idea, Chan [4] describes a metric to estimate the level of user interest; for example the percentage of links visited on a page or URLs presented in bookmarks.

Several systems have attempted to provide personalized search that is based upon user profiles. The Personal Search Assistant [12] acts as an independent agent that collects and organizes information on behalf of its user. The user manually creates a conceptual database that is input to a personal agent responsible for building a user profile. The profile is used to filter the results of later searches. Similarly, the Competitive Intelligence Spider and Meta Spider [5] autonomously gather information for a user based on their preferences. Collected documents are then analyzed and noun phrases are extracted to create a personal dictionary for the user to guide future searches.

In contrast to the above systems, the OBIWAN project [8] focuses on interactive, personalized search rather than background processes. Another major difference is that the user profiles are implicitly created based on browsing histories rather than explicitly created from user input. Search results from a conventional search engine are then classified with respect to the same concept hierarchy used to represent the user profiles. Documents then are re-ranked based upon how well their concepts match those that appear highly weighted in the user profile.

Contextual search is also a research field in which personalization can play an important role. PERSIVAL [14], for example, is a system that provides personalized search on specific medical libraries. Rather than building a user profile, PERSIVAL allows users to augment queries by providing contextual information such as a patient record. PERSIVAL then extracts concepts from the patient records and uses them to expand the query. The patient record is also used to filter the search results. In his thesis, Challam also studied contextual search based on the contents of user desktops [3]. In this work, the contents of any open windows on a user's desktop were captured and classified to create a snapshot of the user's current focus. This profile was submitted to the search engine along with the query and used to promote search results selected by the query that best match the context profile.

3. Approach

Our study investigates the effectiveness of personalized search based upon user profiles built from search histories. The user profiles are represented as weighted concept hierarchies, where the Open Directory Project concept hierarchy [16] is used as the reference concept hierarchy.

In order to capture information about our users, we implemented GoogleWrapper to anonymously monitor the search activities of a set of volunteers. We collected two different types of information per individual user:

1. queries submitted through GoogleWrapper and for which at least one result was visited;
2. snippets (titles and summaries) of results in the list selected by the user.

Each piece of information collected about a user was classified into a concept hierarchy based upon the Open Directory Project [16] hierarchy. For each user, we created and compared two different profiles – one based upon the classified queries (i.e., submitted keywords) and another based upon the classified snippets of the selected results. User profiles are therefore represented as weighted concept hierarchies such that concepts into which more items were classified received higher weights.

Once the user profiles are built, they are used to provide personalized search. After a query is submitted to GoogleWrapper, the search result snippets are

Proceedings of the 2005 IEEE/WIC/ACM International Conference on Web Intelligence (WI'05)
0-7695-2415-X/05 $20.00 © 2005 IEEE

classified into the same reference concept hierarchy. A matching function, described in Section 3.3, calculates the degree of similarity between each result snippets concepts and the user profile concepts. This similarity value, representing a conceptual match, is used to re-rank the search results so that those results that best match the user's interests are higher in the list.

We believe that this approach has several advantages over our previous work [2] in which user profiles were built from user browsing hierarchies. In this work, user interests are collected in a completely non-invasive way, search personalization is based upon data readily available to the search engine, and the system effectiveness can be evaluated by monitoring user activities rather than requiring explicit judgments or feedback. Since this is a server-side technique, we realize that there might be concerns about users' personal information, especially if we consider a large-scale deployment of the technology. However, we believe that user privacy can be protected if we adopt security methodologies that are already widely used such as SSL and access to the service via a login process.

3.1. System Architecture

The architecture of our system consists of two main modules:

1. GoogleWrapper: a wrapper for Google responsible for collecting information from users. Google APIs [11] and nusoap library [19] were used for the implementation. Users register with their email addresses in order to create a cookie to store and upload their userID on their local machines. If the cookie was lost, GoogleWrapper would notify the user and she could login to reset the cookie. When queries are submitted by users, GoogleWrapper stores in a session variable the query and the userID and then forwards the query to the Google search engine [10]. It intercepts the search engine results, stores them into a session variable, re-ranks them, and then displays them to the user. When users click on a result, GoogleWrapper logs the summary and the title of the selected document along with the query and the userID before redirecting the browser to the appropriate Web page. In our deployed system, see http://www.ittc.ku.edu/~mirco/demo, we rerank the results before presenting them to the user. However, during our experiments, GoogleWrapper presented the results of all queries in random order to collect unbiased user feedback.
2. The categorizer from KeyConcept [9], a conceptual search engine, is used to classify each query and snippet into a list of weighted concepts from the reference concept hierarchy. This vector space model classifier implements a k-nearest neighbors algorithm.

A set of scripts was also implemented to conduct the experimental analysis of the effectiveness of using search-history based user profiles for personalized search, comparing Google's original rank with our conceptual rank.

3.2. User Profiles

In this study, user profiles are represented by a weighted concept hierarchy. The reference concept hierarchy contains 1,869 categories in the top 3 levels of the Open Directory Project, and the weights represent the amount of user interest in the specific category. The classifier was trained using 30 documents listed for each category that were collected by a spider. The user profile concept weights are assigned by classifying textual content collected from the user into the appropriate categories. This process produces a list of concepts with an associated weight that can be accumulated over the queries, or snippets, submitted. Essentially, the vocabulary in the submitted queries or snippets is compared with the vocabulary for each category's set of training documents and the classifier reports back a similarity value.

3.3. Personalized Search

During the evaluation phase, each search result (snippet) is classified to create a document profile in the same format as the user profile. The document profile is then compared to the user profile in order to calculate the conceptual similarity between each document and the user's interests. The conceptual match between the document profile and the user profile is calculated using the cosine similarity function.

$$conceptual\_match(user_i, doc_j) = \sum_{k=1}^{N} cwt_{ik} * cwt_{jk}$$

where

cwt_{ik} = Weight of Concept_k in UserProfile_i

cwt_{jk} = Weight of Concept_k in DocumentProfile_j

N = Number of Concepts

The documents are re-ranked by their conceptual similarity to produce the conceptual rank. The final rank of the document is calculated by combining the conceptual rank with Google's original rank using the following weighting scheme:

$$\text{FinalRank} = \alpha*\text{ConceptualRank} + (1-\alpha)*\text{GoogleRank}$$

α has a value between 0 and 1. When α has a value of 0, conceptual rank is not given any weight, and FinalRank is equivalent to the original rank assigned by Google. If α has a value of 1, the search engine ranking is ignored and pure conceptual rank is considered. The conceptual and search engine based rankings can be blended in different proportions by varying the value of α.

4. Experimental Validation

4.1. Experimental Setup

GoogleWrapper, the tool used to monitor user's activities, was used by six volunteers for a period of almost 6 months. These users included faculty members and graduate students from different departments at the University of Kansas, i.e., Electrical Engineering and Computer Science, Mathematics, and Pharmaceutical Chemistry.

4.1.1. Studying User Behavior. Our first version of GoogleWrapper displayed all results retrieved from Google, ten results per page, presented in the original order. We used this version to collect data about how users normally interact with a search engine. After collecting 576 queries for which at least one result was selected, we randomly picked a sample set of 100 queries for detailed analysis. From this sample, we found that 94% of the user-selected results occurred in the first 3 Google-ranked results, and no result after the tenth result, i.e., on the second page, was ever selected. The top-ranked result was by far the most frequently selected (60%), followed by the second (20%), and the third (14%). From these observations we concluded that users rarely, if ever, look for results beyond the first page displayed by the search system. Thus, for our later experiments, we process only the top 10 results retrieved from Google.

Our second conclusion was that users may be influenced by the rank-order of the result presentation. To verify this hypothesis, we modified GoogleWrapper so that it displayed the top ten results from Google in random order. We randomly selected another sample set of 100 queries and performed the same analysis. This time, the original rank of the user-selected results was more uniformly distributed across the 10 results displayed. The top three results as ranked by Google accounted for 46% when results were presented in random order versus 94% when they were shown in the original order. Google's top result was selected only 15% of the time when it was shown in a random position versus 60% when it is presented at the top of the page. From this, we concluded that user judgments are affected by presentation order, so we continued to randomize the search engine results before presenting them to the user in later experiments.

4.1.2. Collecting Sample Data for Personalization Study. The final version of GoogleWrapper, used to collect the user information for our study of personalized search, presented the top 10 results for each query in random order. Using this system, we collected 609 queries for which at least one result was selected. We removed duplicate queries for each user and, from this collection, we selected 47 queries per user (282 total) into the following sets:

- 240 (40 per user) queries were used for training the 2 user profiles (query-based and snippet-based);
- 30 (5 per user) queries were used for testing personalized search parameters;
- 12 (2 per user) queries were used for validating the selected parameters.

In the following section, we present four experiments in which we investigated the effects of user profiles built out of queries and snippets. We measured the accuracy of such profiles by comparing, for user-selected results, Google's original rank with the conceptual rank based on the profile. Because our goal was to evaluate the quality of the user profiles, not produce the best possible search results, we set α to 1 so that Google's original rank did not affect the FinalRank. Once the best conceptual match was determined, we conducted further experiments to evaluate the effect of varying α to produce a final ranking.

4.1.3. Building profile from queries. The user profile was created by categorizing each training query and accumulating the returned concepts and weights. One question that needed to be resolved was, since the categorizer returns a weighted list of concepts, how many of these concepts per query should be used to update the profile. To answer this, we randomly selected 30 queries (5 per user) and performed a detailed analysis of the classification results. For each query, the top 10 concepts returned by the classifier were manually judged as relevant or not. From this, we found that the top 4 concepts assigned per query were relevant 78% of the time, and that the accuracy dropped dramatically further down the list. Thus, in the experiments that follow, we built the query-based profiles from the top 4 classification results per query.

4.1.4. Building profile from snippets. In this case, user-selected snippets (titles and summaries) were used to build the user profiles. As with query-based profiles, we needed to determine the number of classification results to add to the user profile. Once again, we performed an analysis of the classifier accuracy on 30 randomly chosen user-selected snippets (5 per user). Since the snippets contain more text than an average query, the classifier seems to be able to identify more valid concepts per snippet. Compared to the query classification results, the accuracy does not drop as precipitously as we moved down the list of matching concepts. Overall, the top 5 classified results are accurate 71% of the time and, further down the list, the accuracy begins to steadily decrease. Based on this analysis, the snippet-based profiles used in this study were built using the top 5 concepts for each snippet returned by the classifier.

4.2. Experiment 1

The first variable we investigated was the number of training queries necessary to create a profile based upon the query text alone. As mentioned previously, we used the top 4 concepts returned by the classifier for each query. We created user profiles using training sets of 10, 20, 30, and 40 queries. A second variable studied was the number of concepts from the resulting profile to use when calculating the similarity between the profile and the document. We varied this number from 1 through 20. Based on earlier experiments [3], we used the top 7 concepts for each search as the document profile.

The user profiles were evaluated by comparing the conceptual rank to the original rank of the selected result to see if there was any improvement. Although we varied the number of training queries between 10 and 40, we found the best results with 30 training queries and these results are reported here. Figure 1 shows the comparison of Google's original rank to the conceptual rank averaged over all queries as the number of concepts from the user profiles is varied from 1 to 20. The average original Google rank for the 30 testing queries per user is 4.4. In contrast, when 30 training queries are used to build the profile, and 4 concepts from that profile are used during conceptual match, the average conceptual rank is 2.9. Using a paired, two-tailed t-test, this improvement of 33% was found to be statistically significant (p = 0.002).

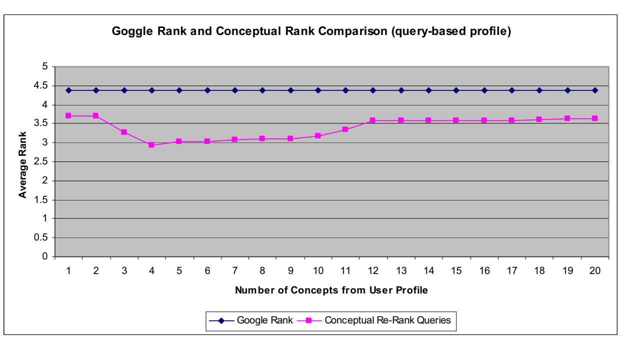

Figure 1: Google's original rank and conceptual rank averaged over all testing queries (user profiles built from queries).

We also examined the results to investigate their effect on the individual testing queries. The query-based profile ranked the selected result higher for 13 queries, hurt the ranking for 3, and left 14 unchanged. Therefore, the re-ranking process helped far more testing queries than it hurt.

4.3. Experiment 2

This experiment repeats Experiment 1, the only difference being that the profiles were built using snippets rather than queries. Once again, conceptual rank was used for evaluation and we used training sets of 10, 20, 30, and 40 snippets. As before, the best results occurred with 30 training snippets.

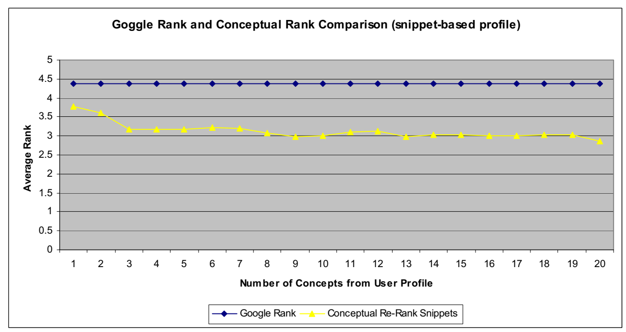

Figure 2: Google's original rank and conceptual rank averaged over all testing queries (user profiles built from snippets).

Fig 2 shows the comparison between Google's rank and the conceptual rank as the number of concepts used from the profile is varied from 1 to 20. The average original Google rank for the 30 testing queries is 4.4. In contrast, when 30 training snippets are used to build the profile, and 20 concepts from the profile are used during conceptual match, the average conceptual rank of the selected result is 2.9. However, the improvement over that when 8 concepts are used is very slight and the curve beyond that point is quite flat showing that the results are somewhat stable beyond that point. Using a paired, two-tailed t-test, this improvement of 34% was found to be statistically significant (p = 0.007). The snippet-based profile improved the ranking for 11 queries, hurt 4, and left 15 unchanged. Once again, the conceptual ranking helped far more queries than it hurt.

4.4. Experiment 3

In this experiment, we looked at combining our conceptual rank with Google's original rank using the FinalRank calculation, described in Section 3.3. Based on the results of Experiment 1, the query-based profile adopted in this experiment, was built using 30 training queries and 4 concepts were used during conceptual ranking. The final rank for a search result was calculated by varying the value of α from 0.0 to 1.0 with a 0.1 step so as to modify the relative contributions of the conceptual and original ranks. Fig 3 shows the average rank comparison between Google's original rank and the FinalRank. The best results are obtained when α is 1.0, i.e., when the original search engine ranking is ignored altogether. This is likely due to the fact that the top 10 results are all good matches for the keywords in the query and, therefore, the distinguishing feature between the results is how well they match the user's interests.

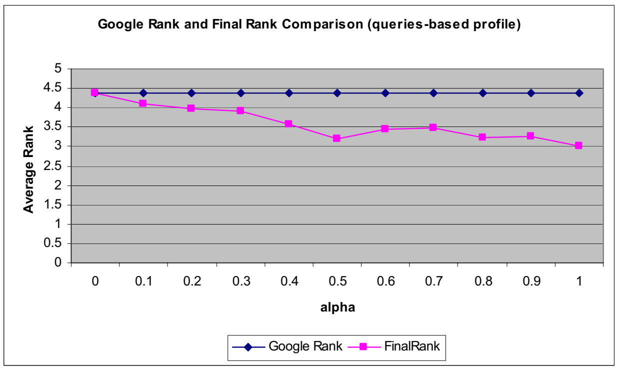

Figure 3: Google's original rank and final rank averaged over all testing queries (user profiles built from queries).

4.5. Experiment 4

Similar to Experiment 3, we examined the effect of combining the search engine's original rank with the conceptual rank. Based on the results of Experiment 2, we used a snippet-based profile built from 30 training snippets and used 20 concepts from the profile for conceptual ranking.

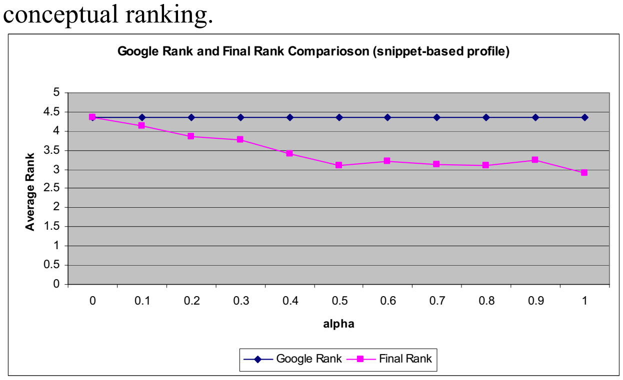

Figure 4: Google's original rank and final rank averaged over all testing queries (user profiles built from snippets).

Figure 4 shows the average rank comparison between Google's original rank and the FinalRank. Once again, the best results occur when α is 1.0, i.e., when the original search engine rankings are ignored altogether.

4.6. Experiment 5

To verify that the user profiles created above are able to improve queries that were not used to tune the profile construction algorithms, we conducted a validation experiment with 12 new testing queries (2 per user). Based on experiments 3 and 4, we compared Google's original rank with the conceptual rank alone.
Table 1 summarizes these results and we notice comparable improvements for the validation queries as observed for the original test queries used to tune the profile creation algorithms. The average rank of the user selected result was 4.8 according to Google, 3.5 based on conceptual match with the snippet-based profile, and 1.8 when the query-based profile was used. The query-based profile produced a 37% improvement and the snippet-based profile produced a 27% improvement on the validation queries.

5. Conclusion and future work

We built a system that creates two types of user profiles based on implicitly collected information: queries submitted and snippets of user-selected results. Using the queries and/or snippets collected by the search engines as the basis for user profiling has several advantages. The information is readily available to the search engines and users do not have to install software or use a proxy server so that browsing or other activities can be monitored. We were able to demonstrate that this information is sufficient to provide significantly improved personalized rankings. We found that using a profile built from 30 queries produced an improvement of 33% in the rank of the selected result. A user profile built from snippets of 30 user-selected results showed an equivalent improvement of 34%. From this we conclude that, although the text users provide to search engines is quite short, it sufficient to provide personalized results. The user profiles we used to build were based on a three-level deep concept hierarchy. We would like to examine the effect of varying the number of levels of the concept hierarchy used in our user profiles. Also, the current concept hierarchy is static, and we would like to evaluate algorithms to dynamically adapt the hierarchy for specific users by merging and/or splitting concepts based upon the amount of user interest.

Table 1. Conceptual rank of user-selected results for the set of 12 validation queries.

Ranking Algorithm	Avg. Rank	Improvement	# Training Queries/Snippets	# Profile Concepts Used
Google (Original)	4.8	--	--	--
Query-based Profile	1.8	37%	30	4
Snippet-based	3.5	27%	30	20

Finally, one challenge for any system that attempts to provide personalized information is to balance the desire to provide better results for users and the need to protect their privacy.

6. References

[1] D. Billsus, M.J. Pazzani. A hybrid user model for news story classification. In proceedings of the seventh international conference on User modeling, Banff, Canada, pp. 99 – 108, 1999.

[2] J. Chaffee, S. Gauch. Personal Ontologies for Web Navigation. In Proceedings of the 9th International Conference on Information and Knowledge Management. (CIKM), pp 227 – 234, 2000.

[3] V. Challam. Contextual information retrieval using ontology based user profile. Master's thesis, University of Kansas, Lawrence, KS, 2004.

[4] P.K. Chan. A Non-Invasive Learning Approach to Building Web User Profiles. KDD-99 Workshop on web usage analysis and user profiling, pp. 7 – 12, 1999.

[5] M. Chau, D. Zeng, H. Chen. Personalized spiders for web search and analysis. In Proceedings of the 1st ACM-IEEE Joint Conference on Digital Libraries, pp 79 - 87, 2001.

[6] C.C. Chen, M.C. Chen, Y. Sun. PVA: a self-adaptive personal view agent. In proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining 2001, San Francisco, California, pp. 257 – 262, 2001.

[7] M. Claypool , P. Le, M. Waseda, D. Brown. Implicit Interest Indicators. Proceedings of the 6th international conference on Intelligent user interfaces (ACM), pp 33 - 40, 2001.

[8] S. Gauch, J. Chafee and A. Pretschner. Ontology-Based Personalized Search and Browsing, Web Intelligence and Agent Systems. Vol. 1 No. 3-4, April 2004, pp.219-234.

[9] S. Gauch, D. Ravindran, S. Induri, J. Madrid, and S. Chadalavada, Internal Technical Report ITTC-FY2004-TR-8646-37, Information and Telecommunication Technology Center, University of Kansas

[10] Google Search Engine. http://www.google.com.

[11] Google APIs. http://www.google.com/apis.

[12] P.R. Kaushik, K. Narayana Murthy. Personal Search Assistant: a configurable personal meta search engine. Fifth Australian World Wide Web Conference, Southern Cross University, Lismore, Australia, 1999.

[13] H.R. Kim, P.K. Chan. Learning implicit user interest hierarchy for context in personalization. In Proceedings of the 8th international conference on Intelligent user interfaces, Miami, Florida, USA, 2003, pp. 101 - 108.

[14] K.R. McKeown, S. Chang, J. Cimino, S.K. Feiner, C. Friedman, L. Gravano, V. Hatzivassiloglou, S. Jhonson, D.A. Jordan, J.L.Klavans, A. Kushniruk, V. Patel and S. Teufel. PERSIVAL, a system for personalized search and summarization over multimedia healthcare information. Proceedings of the first ACM/IEEE-CS joint conference on Digital libraries, pp 331 – 340, 2001.

[15] S. Mizzaro, C. Tasso. Ephemeral and persistent personalization in adaptive information access to scholarly publications on the web. Lecture Notes in Computer Science. In proceedings of the Second International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems, pp 306 – 316, 2002.

[16] The Open Directory Project (ODP). http://dmoz.org.

[17] OnStat.com. Most people use 2 word phrases in search engines according to OneStat.com. http://www.onstat.com, Feb 2004.

[18] M. Pazzani, J. Muramatsu and D. Billsus. Syskill & Webert: identifying interesting Web sites. In proceedings of the 13th National Conference on Artificial Intelligence (AAAI96), pp 54 – 61, 1996.

[19] PHP Library for Google APIs. http://dietrich.ganx4.com/nusoap/.

[20] StatMarket.com. Search guiding more web activity. http://www.statmarket.com, 2003

[21] J. Trajkova and S. Gauch, Improving Ontology-Based User Profiles. RIAO 2004, Vaucluse, France, April 26-28, pp. 380-389.