International Journal of Modern Engineering Research (IJMER)
www.ijmer.com Vol.2, Issue.1, Jan-Feb 2012 pp-260-264 ISSN: 2249-6645

# RANKING CONCEPT-BASED USER PROFILE FROM SEARCH ENGINE LOGS

**R.Kokila[1] and Mrs.N.Krishnammal[2]**

[1]M.E Computer Science and Engineering, Sri Shakthi Institute of Engineering and Technology, Coimbatore, India
[2]Assistant Professor, Department of Computer Science and Engineering, Sri Shakthi Institute of Engineering and Technology, Coimbatore, India

**ABSTRACT:**

Commercial search engines return roughly the same results for the same query, regardless of the user's real interest. Since queries submitted to search engines tend to be short and ambiguous, they are not likely to be able to express the user's precise needs. In existing system, most existing user profiling strategies are based on objects that users are interested in (i.e., positive preferences), but not the objects that users dislike (i.e., negative preferences). Experimental results show that profiles which capture and utilize both of the user's positive and negative preferences perform the best and also negative preferences can increase the separation between similar and dissimilar queries. The separation can be achieved by using agglomerative clustering algorithm to terminate and improve the overall quality of the resulting query clusters. In the proposing system, queries submitted to search engines, they are likely to be able to express the user's precise needs and the concept based user profiles can be integrated into the ranking algorithms of a search engine so that search results can be ranked according to individual user's interests. This technique improves a search engine's performance by identifying the information needs for individual users.

**Index terms-** Negative preferences, personalization, personalized query clustering, search engine, user profile.

## I.INTRODUCTION

MOST commercial search engines return roughly the same results for the same query, regardless of the user's real interest. Since queries submitted to search engines tend to be short and ambiguous, they are not likely to be able to express the user's precise needs. For example, a farmer may use the query "apple" to find information about growing delicious apples, while graphic designers may use the same query to find information about Apple Computer.

Personalized search is an important research area that aims to resolve the ambiguity of query terms. To increase the relevance of search results, personalized search engines create user profiles to capture the users' personal preferences and as such identify the actual goal of the input query. Since users are usually reluctant to explicitly provide their preferences due to the extra manual effort involved, recent research has focused on the automatic learning of user preferences from users' search histories or browsed documents and the development of personalized systems based on the learned user preferences.

A good user profiling strategy is an essential and fundamental component in search engine personalization. We studied various user profiling strategies for search engine personalization, and observed the following problems in existing strategies.

Most personalization methods focused on the creation of one single profile for a user and applied the same profile to all of the user's queries. We believe that different queries from a user should be handled differently because a user's preferences may vary across queries. For example, a user who prefers information about fruit on the query "orange" may prefer the information about Apple Computer for the query "apple."

Existing click through-based user profiling strategies can be categorized into document-based and concept based approaches. They both assume that user clicks can be used to infer users' interests, although their inference methods and the outcomes of the inference are different. Document-based profiling methods try to estimate users' document preferences (i.e., users are interested in some documents more than others). On the other hand, concept based profiling methods aim to derive topics or concepts that users are highly interested. Document-based methods that consider both users' positive and negative preferences, to the best of our knowledge, there are no concept-based methods that considered both positive and negative preferences in deriving user's topical interests.

Most existing user profiling strategies only consider documents that users are interested in (i.e., users' positive preferences) but ignore documents that user's dislike (i.e., users' negative preferences). For example, if a user is interested in "apple" as a fruit, he/she may be interested specifically in apple recipes, but less interested in information about growing apples, while absolutely not interested in information about the company Apple Computer. In this case, a good user profile should favor information about apple recipes, slightly favor information about growing apple, while downgrade information about Apple Computer. Profiles built on both positive and negative user preferences can represent user interests at finer details.

The main contributions of this paper are:

- The query-oriented, concept-based user profiling method [11] to consider both users' positive and negative preferences in building users profiles. We proposed six user profiling methods that exploit a user's positive and negative preferences to produce a profile for the user using a Ranking SVM (RSVM).
- While document-based user profiling methods pioneered by Joachim's [10] capture users' document preferences (i.e., users consider some documents to be more relevant than others), our methods are based on users' concept preferences (i.e., users consider some topics/concepts to be more relevant than others).
- RSVM to learn from concept preferences weighted concept vectors representing concept-based user profiles. The weights of the vector elements, which could be positive or negative, represent the interestingness (or uninteresting ness) of the user on the concepts [11]. The weights that represent a user's interests are all positive, meaning that the method can only capture user's positive preferences.

## II.AN OVERVIEW OF RELARED WORK

User profiling strategies can be broadly classified into two main approaches: document-based and concept-based approaches. Document-based user profiling methods aim at capturing users' clicking and browsing behaviors. Users' document preferences are first extracted from the click through data, and then, used to learn the user behavior model which is usually represented as a set of weighted features. On the other hand, concept-based user profiling methods aim at capturing users' conceptual needs. Users' browsed documents and search histories are automatically mapped into a set of topical categories. User profiles are created based on the users' preferences on the extracted topical categories.

### A. Document-Based Methods

Most document-based methods focus on analyzing users' clicking and browsing behaviors recorded in the users' click through data. On Web search engines, click through data are important implicit feedback mechanism from users. An example of click through data for the query "apple," which contains a list of ranked search results presented to the user, with identification on the results that the user has clicked on. Several personalized systems that employ click through data to capture users' interest have been proposed [1], [2], [10].

### B.Concept-Based Methods

Most concept-based methods automatically derive users' topical interests by exploring the contents of the users' browsed documents and search histories. Liu et al. [13] proposed a user profiling method based on users' search history and the Open Directory Project (ODP) [16]. The user profile is represented as a set of categories, and for each category, a set of keywords with weights. The categories stored in the user profiles serve as a context to disambiguate user queries. If a profile shows that a user is interested in certain categories, the search can be narrowed down by providing suggested results according to the user's preferred categories.

## III. PERSONALIZED CONCEPT-BASED QUERY CLUSTERING

Our personalized concept-based clustering method consists of three steps. First, concept extraction algorithm, extract concepts and their relations from the Web-snippets returned by the search engine. Second, seven different concept-based user profiling strategies, to create concept based user profiles. Finally, the concept-based user profiles are compared with each other and against as baseline our previously proposed personalized concept-based clustering algorithm.

### A. Concept Extraction

After a query is submitted to a search engine, a list of Web snippets is returned to the user. We assume that if a keyword/phrase exists frequently in the Web-snippets of a particular query, it represents an important concept related to the query because it coexists in close proximity with the query in the top documents. Thus, we employ the following support formula, which is inspired by the well-known problem of finding frequent item sets in data mining [7], to measure the interestingness of a particular keyword/phrase extracted from the Web-snippets.

B. Query Clustering Algorithm

Concept-based clustering algorithm with which ambiguous queries can be classified into different query clusters. Concept-based user profiles are employed in the clustering process to achieve personalization effect. First, a query-concept bipartite graph G is constructed by the clustering algorithm in which one set of nodes corresponds to the set of users' queries and the other corresponds to the sets of extracted concepts. Each individual query submitted by each user is treated as an individual node in the bipartite graph by labeling each query with a user identifier. Concepts with interestingness weights greater than zero in the user profile are linked to the query with the corresponding interestingness weight in G. Second, a two-step personalized clustering algorithm is applied to the bipartite graph G, to obtain clusters of similar queries and similar concepts.

$$sim(x,y) = \frac{N_x \cdot N_y}{\| N_x \| \| N_y \|}, \qquad (7)$$

where Nx is a weight vector for the set of neighbor nodes of node x in the bipartite graph G, the weight of a neighbor node nx in the weight vector Nx is the weight of the link connecting x and nx in G, Ny is a weight vector for the set

of neighbor nodes of node y in G, and the weight of a neighbor node ny in Ny is the weight of the link connecting y and ny in G.

Algorithm 1. Personalized Agglomerative Clustering

Input: A Query-Concept Bipartite Graph G

Output: A Personalized Clustered Query-Concept Bipartite

Graph Gp

// Initial Clustering

1: Obtain the similarity scores in G for all possible pairs of query nodes using Equation (7).

2: Merge the pair of most similar query nodes (qi,qj) that does not contain the same query from different users. Assume that a concept node c is connected to both query nodes qi and qj with weight wi and wj, a new link is created between c and (qi; qj) with weight w = wi +wj.

3: Obtain the similarity scores in G for all possible pair's of concept nodes using Equation (7).

4: Merge the pair of concept nodes (ci,cj) having highest similarity score. Assume that a query node q is connected to both concept nodes ci and cj with weight wi and wj, a new link is created between q and (ci; cj) with weight w = wi +wj.

5. Unless termination is reached, repeat Steps 1-4.
// Community Merging

6. Obtain the similarity scores in G for all possible pairs of query nodes using Equation (7).

7. Merge the pair of most similar query nodes (qi,qj) that contains the same query from different users. Assume that a concept node c is connected to both query nodes qi and qj with weight wi and wj, a new link is created between c and (qi; qj) with weight w = wi + wj.

8. Unless termination is reached, repeat Steps 6-7.

## IV. USER PROFILE STRATEGIES

Six user profiling strategies which are both concept-based and utilize users' positive and negative preferences. They are PJoachims_C, PmJoachims_C, PSpyNB_C, PClickþJoachims_C, PClickþmJoachims_C, and PClickþSpyNB_C.

**A.Click-Based Method (PClick)**

The concepts extracted for a query q using the concept extraction method the possible concept space arising from the query q. The concept Space may cover more than what the user actually wants. For example, when the user searches for the query "apple," the concept space derived from our concept extraction method contains the concepts "macintosh," "ipod," and "fruit." If the user is indeed interested in "apple" as a fruit and clicks on pages containing the concept "fruit," the user Profile represented as a weighted concept vector should record the user interest on the concept "apple" and its neighborhood (i.e., concepts which having similar meaning as "fruit"), while downgrading unrelated concepts such as "macintosh," "ipod," and their neighborhood.

**B.Joachims-C Method (PJoachims_C)**

Given a list of search results for an input query q, if a user clicks on the document dj at rank j, all the concepts C(di) in the unclicked documents di above rank j are considered as less relevant than the concepts C(dj) in the document dj, i.e., (C(dj) <r' C(di)), where r' is the user's preference order of the concepts extracted from the search results of the query q).

**C. mJoachims-*C Method (PmJoachims_C)***

Given a set of search results for a query, if document di at rank i is clicked, dj is the next clicked document right after di (no other clicked links between di and dj), and document dk at rank k between di and dj (i < k < j) is not clicked, then conceptsC(dk) in document dk is considered less relevant than the concepts C(dj) in document dj (C(dj) <r' C(dk)), where r' is the user's preference order of the concepts extracted from the search results of the query q).

*D.SpyNB-C Method (PSpyNB_C)*

Both Joachims and mJoachims are based on a rather strong assumption that pages scanned but not clicked by the user are considered uninteresting to the user, and hence, irrelevant to the user's query. the search engine context, most users would only click on a few documents (positive examples) that are relevant to them. Thus, only a limited number of positive examples can be used in the classification process, lowering the reliability of the predicted negative examples.

**E.Click+Joachims-C Method (PClickþJoachims_C)**

Integrate the click-based method, which captures only positive preferences, with the Joachims-C method, with which negative preferences can be obtained. We found that Joachims-C is good in predicting users' negative preferences. Since both the user profiles PClick and PJoachims_C are represented as weighted concept vectors, the two vectors can be combined.

**F.Click+mJoachims-C Method (PClickþmJoachims_C)**

Similar to Click+Joachims-C method, a hybrid method which combines PClick and PmJoachims_C is proposed.

**G. *Click+SpyNB-C Method (PClickþSpyNB_C)***

Similar to Click+Joachims-C and Click+mJoachims-C methods.

## V. EXPERIMENTAL RESULTS

We evaluate and analyze the seven conceptbased user profiling strategies (i.e., PClick, PJoachims_C, PmJoachims_C, PSpyNB_C, PClick+Joachims_C, PClick+mJoachims_C, and PClick+SpyNB_C). The seven concept-based user profiling strategies are compared using our personalized concept-based clustering algorithm [11]. The collected clickthrough data are used by the proposed user profiling strategies to create user profiles. The performance of a heuristic for determining the termination points of initial clustering and community merging based on the change of intracluster similarity. We show that user profiling methods that incorporate negative concept weights return termination points that are very close to the optimal points obtained by exhaustive search.

### A. Experimental Setup

The query and click through data for evaluation are adopted from our previous work [11]. To evaluate the performance of our user profiling strategies, we developed a middleware for Google to collect click through data. We used 500 test queries, which are intentionally designed to have ambiguous. The clusters obtained from the algorithms are compared against the standard clusters to check for their correctness. The 100 users are invited to use our middleware to search for the answers of the 500 test queries (accessible at [3]). To avoid any bias, the test queries are randomly selected from 10 different categories.

The user profiles are employed by the personalized clustering method to group similar queries together according to users' needs. The personalized clustering algorithm is a two-phase algorithm which composes of the initial clustering phase to cluster queries within the scope of each user, and then, the community merging phase to group queries for the community.

### B.Comparing Concept Preference Pairs Obtained Using Joachims-C, mJoachims-C, and SpyNB-C Methods

In this section, we evaluate the pair wise agreement between the concept preferences extracted using Joachims-C, mJoachims- C, and SpyNB-C methods. The three methods are employed to learn the concept preference pairs from the collected click through data. The learned concept preference pairs from different methods are manually evaluated by human evaluators to derive the fraction of correct preference pairs. We discard all the ties in the resulted concept preference pairs (i.e., pairs with the same concepts) to avoid ambiguity (i.e., both ci > cj and cj > ci exist) in the evaluation. RSVM is then employed to learn user profiles from the concept preference pairs.

### C.Comparing PClick, PJoachims_C, PmJoachims_C, PSpyNB_C, PClick+Joachims_C, PClick+mJoachims_C, and PClick+SpyNB_C

An important observation from these three figures is that even though PJoachims_C, PmJoachims_C, and PSpyNB_C are able to capture users' negative preferences, they yield worse precision and recall ratings comparing to PClick. This is attributed to the fact that PJoachims_C, PmJoachims_C, and PSpyNB_C share a common deficiency in capturing users' positive preferences. A few wrong positive predictions would significantly lower the weight of a positive concept.

For example, assume that a positive concept ci has been clicked many times, a preference cj <r0 ci can still be generated by Joachims/mJoachims propositions, if there ever exists one case in which the user did not click on ci but clicked on another document that was ranked lower in the result list. Since PJoachims_C, PmJoachims_C, and PSpyNB_C cannot effectively capture users' positive preferences, they perform worse than the baseline method PClick. On the other hand, PClick captures positive preferences based on user clicks, so an erroneous click made by users has little effect on the final outcome as long as the number of erroneous clicks is much less than that of correct clicks.

### D.Termination Points for Individual Clustering to Community Merging

As initial clustering is run, a tree of clusters will be built along the clustering process. The termination point for initial clustering can be determined by finding the point at which the cluster quality has reached its highest (i.e., further clustering steps would decrease the quality). The same can be done for determining the termination point for community merging.

Fig. 1. Change in similarity values when performing personalized clustering using PClick+Joachims_C.

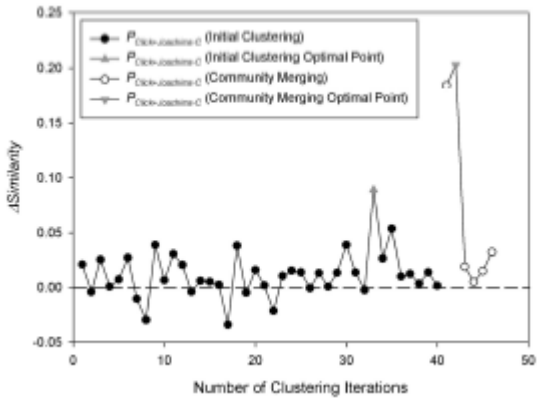
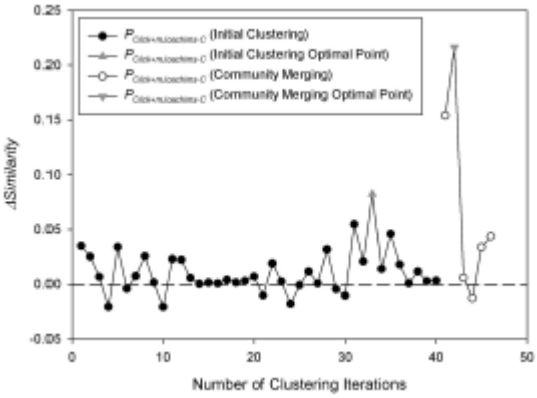


Fig. 2. Change in similarity values when performing personalized clustering using PClick+mJoachims_C.




## VI. CONCLUSION

An accurate user profile can greatly improve a search engine's performance by identifying the information needs for individual users. In this paper, we proposed and evaluated several user profiling strategies. The techniques make use of click through data to extract from Web-snippets to build concept-based user profiles automatically. We applied preference mining rules to infer not only users' positive preferences but also their negative preferences, and utilized both kinds of preferences in deriving user's profiles. The user profiling strategies were evaluated and compared with the personalized query clustering method that we proposed previously. Our experimental results show that profiles capturing both of the user's positive and negative preferences perform the best among the user profiling strategies studied. Apart from improving the quality of the resulting clusters, the negative preferences in the proposed user profiles also help to separate similar and dissimilar queries into distant clusters, which help to determine near optimal terminating points for our clustering algorithm.

We plan to take on the following two directions for future work. First, relationships between users can be mined from the concept-based user profiles to perform collaborative filtering. This allows users with the same interests to share their profiles. Second, the existing user profiles can be used to predict the intent of unseen queries, such that when a user submits a new query, personalization can benefit the unseen query. Finally, the concept-based user profiles can be integrated into the ranking algorithms of a search engine so that search results can be ranked according to individual users' interests.

## REFERENCES

[1] E. Agichtein, E. Brill, and S. Dumais, "Improving Web Search Ranking by Incorporating User Behavior Information," Proc. ACM SIGIR, 2006.

[2] E. Agichtein, E. Brill, S. Dumais, and R. Ragno, "Learning User Interaction Models for Predicting Web Search Result Preferences," Proc. ACM SIGIR, 2006.

[3] Appendix: 500 Test Queries, http://www.cse.ust.hk/~dlee/ tkde09/Appendix.pdf, 2009.

[4] R. Baeza-yates, C. Hurtado, and M. Mendoza, "Query Recommendation Using Query Logs in Search Engines," Proc. Int'l Workshop Current Trends in Database Technology, pp. 588-596, 2004.

[5] D. Beeferman and A. Berger, "Agglomerative Clustering of a Search Engine Query Log," Proc. ACM SIGKDD, 2000.

[6] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender, "Learning to Rank Using Gradient Descent," Proc. Int'l Conf. Machine learning (ICML), 2005.

[7] K.W. Church, W. Gale, P. Hanks, and D. Hindle, "Using Statistics in Lexical Analysis," Lexical Acquisition: Exploiting On-Line Resources to Build a Lexicon, Lawrence Erlbaum, 1991.

[8] Z. Dou, R. Song, and J.-R. Wen, "A Largescale Evaluation and Analysis of Personalized Search Strategies," Proc. World Wide Web (WWW) Conf., 2007.

[9] S. Gauch, J. Chaffee, and A. Pretschner, "Ontology-Based Personalized Search and Browsing," ACM Web Intelligence and Agent System, vol. 1, nos. 3/4, pp. 219-234, 2003.

[10] T. Joachims, "Optimizing Search Engines Using Clickthrough Data," Proc. ACM SIGKDD, 2002.

[11] K.W.-T. Leung, W. Ng, and D.L. Lee, "Personalized Concept-Based Clustering of Search Engine Queries," IEEE Trans. Knowledge and Data Eng., vol. 20, no. 11, pp. 1505-1518, Nov. 2008.

[12] B. Liu, W.S. Lee, P.S. Yu, and X. Li, "Partially Supervised Classification of Text Documents," Proc. Int'l Conf. Machine Learning (ICML), 2002.

[13] F. Liu, C. Yu, and W. Meng, "Personalized Web Search by Mapping User Queries to Categories," Proc. Int'l Conf. Information and Knowledge Management (CIKM), 2002.

[14] Magellan, http://magellan.mckinley.com/, 2008.

[15] W. Ng, L. Deng, and D.L. Lee, "Mining User Preference Using Spy Voting for Search Engine Personalization," ACM Trans. Internet Technology, vol. 7, no. 4, article 19, 2007.

[16] Open Directory Project, http://www.dmoz.org/, 2009.

[17] M. Speretta and S. Gauch, "Personalized Search Based on User Search Histories," Proc. IEEE/WIC/ACM Int'l Conf. Web Intelligence, 2005.

[18] Q. Tan, X. Chai, W. Ng, and D. Lee, "Applying Co-training to Clickthrough Data for Search Engine Adaptation," Proc. Database Systems for Advanced Applications (DASFAA) Conf., 2004.

[19] J.-R. Wen, J.-Y. Nie, and H.-J. Zhang, "Query Clustering Using User Logs," ACM Trans. Information Systems, vol. 20, no. 1, pp. 59- 81, 2002.

[20] Y. Xu, K. Wang, B. Zhang, and Z. Chen, "Privacy-Enhancing Personalized Web Search," Proc. World Wide Web (WWW) Conf., 2007.

## Authors

**Mr.Kokila.R** received B.E degree in CSE from Anna University, Chennai and Currently pursuing M.E degree in Computer Science and Engineering in Sri Shakthi Institute of Engineering and Technology, under Anna University of Technology, Coimbatore. Her research interest Includes Computer Networks and Data Mining.

**Mrs.N.Krishnammal** received the B.E degree in ECE from Anna University, Chennai and Received the M.E degree in CSE from Anna University of technology, Coimbatore and pursuing Phd in Networks under Anna university of technology, Coimbatore. She is currently working as Assistant Professor in Department of CSE in Sri Shakthi Institute of Engineering and Technology, Coimbatore. Her main research interest is Computer Networks.