ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 8, August 2014

Collaborative Filtering Based On Search Engine Logs

K.SIVARAMAN[1]

Assistant Professor, Dept. of Computer Science, BIST, Bharath University, Chennai -600073[1]

ABSTRACT: Search engines return roughly the same results for the same query, regardless of the user's real interest. 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. In this project, we focus on search engine personalization and develop several concept-based user profiling methods that are based on both positive and negative preferences. User profiles which capture both the user's positive and negative preferences. Negative preferences improve the separation of similar and dissimilar queries, which facilitates an agglomerative clustering algorithm to decide if the optimal clusters have been obtained.

KEYWORDS: Negative preferences ,personalization , agglomerative clustering algorithm, search engine, user profiling

I. INTRODUCTION

Data mining is often defined as finding hidden information in a database. Data mining is classified into two types predictive and descriptive. predictive model makes a prediction about values of data using known results found from different data. A descriptive model identifies patterns or relationships in data clustering comes under the category of descriptive. Clustering is classified into hierarchical, partitional, categorical, large database. A hierarchical algorithm creates a set of clusters. Hierarchical algorithms are classified into two types, agglomerative algorithm and divisive algorithm. In this agglomerative clustering algorithm concept is used to cluster the similar query and similar concepts to obtain the optimal results of clusters.

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. In this research, we address the

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 8, August 2014

above problems by proposing and studying seven concept-based user profiling strategies that are capable of deriving both of the user's positive and negative preferences. The entire user profiling strategies is query oriented, meaning that a profile is created for each of the user's queries. The user profiling strategies are evaluated and compared with our previously proposed personalized query clustering method.

The user profiles which capture both the user's positive and negative preferences perform the best among all of the profiling strategies studied. Moreover, we find that negative preferences improve the separation of similar and dissimilar queries, which facilitates an agglomerative clustering algorithm to decide if the optimal clusters have been obtained.

The organization of this paper is as follows: Section 2 describes a background of this paper, section 3 explain the personalized agglomerative algorithm, section 4 described methods of problem and selection 5 and 6 are given conclusion and references.

II. BACKGROUND

A major problem of current Web search is that search queries are usually short and ambiguous, and thus are insufficient for specifying the precise user needs. To alleviate this problem, some search engines suggest terms that are semantically related to the submitted queries so that users can choose from the suggestions the ones that reflect their information needs. In this paper, we introduce an effective approach that captures the user's conceptual preferences in order to provide personalized query suggestions. We achieve this goal with two new strategies. First, we develop online techniques that extract concepts from the web-snippets of the search result returned from a query and use the concepts to identify related queries for that query. Second, we propose a new two phase personalized agglomerative clustering algorithm that is able to generate personalized query clusters. To the best of the author's knowledge, no previous work has addressed personalization for query suggestions. To evaluate the effectiveness of our technique, a Google middleware was developed for collecting click through data to conduct experimental evaluation. Experimental results show that our approach has better precision and recall than the existing query clustering methods.[1]

User profiles, descriptions of user interests, can be used by search engines to provide personalized search results. Many approaches to creating user profiles capture user information through proxy servers (to capture browsing histories) or desktop bots (to capture all activities on a personal computer). These both 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. In our study, we implemented a wrapper for Google to examine different sources of information on which to base the user profiles: queries and snippets of examined search results. These user profiles were created by classifying the information into concepts from the Open Directory Project concept hierarchy and then used to re-rank the search results. User feedback was collected to compare Google's original rank with our new rank for the results examined by users. We found that queries were as effective as snippets when used to create user profiles and that our personalized re-ranking resulted in a 37% improvement in the rank-order of the user-selected results.[2]

An approach to automatically optimizing the retrieval quality of search engines using click through data. Intuitively, a good information retrieval system should present relevant documents high in the ranking, with less relevant documents following below. While previous approaches to learning retrieval functions from examples exist, they typically require training data generated from relevance judgments by experts. This makes them difficult and expensive to apply. The goal of this paper is to develop a method that utilizes click through data for training, namely the query-log of the search engine in connection with the log of links the users clicked on in the presented ranking. Such click through data is available in abundance and can be recorded at very low cost. Taking a Support Vector Machine (SVM) approach, this paper presents a method for learning retrieval functions. From a theoretical

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 8, August 2014

perspective, this method is shown to be well-founded in a risk minimization framework. Furthermore, it is shown to be feasible even for large sets of queries and features. The theoretical results are verified in a controlled experiment. It shows that the method can effectively adapt the retrieval function of a meta-search engine to a particular group of users, outperforming Google in terms of retrieval quality after only a couple of hundred training example.[3]

Query clustering is a process used to discover frequently asked questions or most popular topics on a search engine. This process is crucial for search engines based on question-answering. Because of the short lengths of queries, approaches based on keywords are not suitable for query clustering. This paper describes a new query clustering method that makes use of user logs which allow us to identify the documents the users have selected for a query. The similarity between two queries may be deduced from the common documents the users selected for them. Our experiments show that a combination of both keywords and user logs is better than using either method alone.[4]

Mining a collection of user transactions with an Internet search engine to discover clusters of similar queries and similar URLs. The information we exploit is "click through data":each record consists of a user's query to a search engine along with the URL which the user selected from among the candidates offered by the search engine. By viewing this dataset as a bipartite graph, with the vertices on one side corresponding to queries and on the other side to URLs, one can apply an agglomerative clustering algorithm to the graph's vertices to identify related queries and URLs. One noteworthy feature of the proposed algorithm makes no use of the actual content of the queries or URLs, but only how they co-occur within the click through data.[5] Clustering is a data mining technique which is used to determine the similarity among the data on predefined attributes. The most similar data are grouped as clusters.

But in this proposed work, we focus on search engine personalization and develop several concept-based user profiling methods that are based on both positive and negative preferences. Negative preferences improve the separation of similar and dissimilar queries, which facilitates an agglomerative clustering algorithm to decide if the optimal clusters have been obtained.

Proposed methods use an RSVM to learn from concept preferences weighted concept vectors representing concept-based user profiles.

III. ALGORITHM FOR PERSONALIZED AGGLOMERATIVE CLUSTERING

The personalized clustering algorithm iteratively merges the most similar pair of query nodes, and then, the most similar pair of concept nodes, and then, merge the most similar pair of query nodes, and so on. The following cosine similarity function is employed to compute the similarity score sim(x,y) of a pair of query nodes or a pair of concept nodes.

$$sim(x,y) = Nx \cdot Ny / || Nx || || Ny || \quad \text{eqn---(1)}$$

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 : 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(1)

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 8, August 2014

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 pairs of concept nodes using Equation (1).
   4. Merge the pair of concept nodes (ci,cj) having highest similarity score.

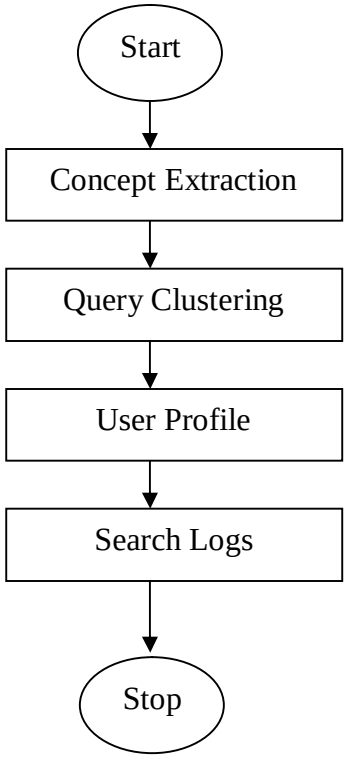
Assume that a query node q is connected to both concept nodes ci and cj with weight wi and wj, a new link is created between q and (ci,cj) with weight w=wi+wj

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 (1).
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. ARCHITECTURE OF THE PROPOSED SYSTEM



METHOD DESCRIPTION:

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 8, August 2014

The Personalized Agglomerative Clustering algorithm is divided into two steps: initial clustering and community merging.

4.1 Initial Clustering:

In initial clustering, queries are grouped within the scope of each user. And the initial clustering is involved in the Personalized Agglomerative Clustering algorithm.

4.2 Community Merging:

Community merging is then involved to group queries for the community. And the Community merging is involved in the Personalized Agglomerative Clustering algorithm.

4..3 Termination point:

A common requirement of iterative clustering algorithms is to determine when the clustering process should stop to avoid over merging of the clusters. When the termination point for initial clustering is reached, community merging kicks off; when the termination point for community merging is reached, the whole algorithm terminates. Good timing to stop the two phases is important to the algorithm, since if initial clustering is stopped too early (i.e., not all clusters are well formed), community merging merges all the identical queries from different users, and thus, generates a single big cluster without much personalization effect. If initial clustering is stopped too late, the clusters are already overly merged before community merging begins. The low precision rate thus resulted would undermine the quality of the whole clustering process.

The termination point form 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. The change in cluster quality can be measured by ΔSimilarity, which is the change in the

similarity value of the two most similar clusters in two consecutive steps. For efficiency reason, we adopt the single-link approach to measure cluster similarity. The similarity of two cluster is the same as the similarity between the two most similar queries across the two clusters.

Formally, ΔSimilarity is defined as

ΔSimilarity(i)= simi(Pqm,Pqn) _ simi+1(Pqo,Pqp)

where qm and qn are the two most similar queries in the ith step of the clustering process, P(qm)and P(qn) are the concept-based profiles for qm and qn, qo and qp are the two most similar queries in the (i+1)th step of the clustering process, P(qo) and P(qp) are the concept-based profiles for qm and qn, and sim() is the cosine similarity.

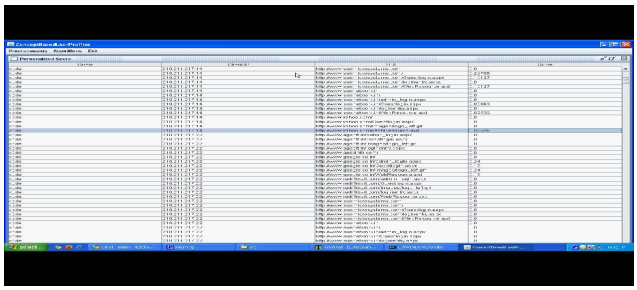


Fig 1.Conceept based used profile

ISSN(Online): 2320-9801
ISSN (Print): 2320-9798

International Journal of Innovative Research in Computer and Communication Engineering

(An ISO 3297: 2007 Certified Organization)

Vol. 2, Issue 8, August 2014

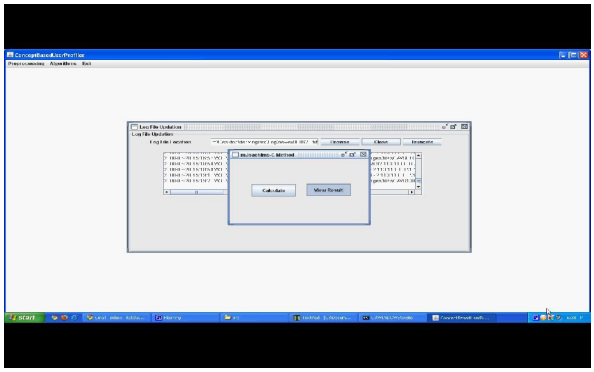


Fig 2: Log based profile

V. CONCLUSION AND FUTURE WORK

An accurate user profile can greatly improve a search engine's performance by identifying the information needs for individual users. 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 user's 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.

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 helps to determine near optimal terminating points for our clustering algorithm.

We observe that the algorithmic optimal points for initial clustering and community merging usually are only one step away from the manually determined optimal points. Further, the precision and recall values obtained at the algorithmic optimal points are only slightly lower than those obtained at the manually determined optimal points.

In the future work, 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.

REFERENCES

[1] 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.
[2] M. Speretta and S. Gauch, "Personalized Search Based on User Search Histories," Proc. EEE/WIC/ACM Int'l Conf. Web Intelligence, 2005.
[3] T. Joachim's, "Optimizing Search Engines Using Click through Data," Proc. ACM SIGKDD, 2002.
[4] 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.
[5] D. Beeferman and A. Berger, "Agglomerative Clustering of a Search Engine Query Log," Proc. ACM SIGKDD, 2000.