Securing Privacy in Profile-based Personalized Web Search

Manali Wadnerkar[1] Dr. D.R.Ingle[2]
[1,2]Department of Computer Engineering
[1,2]University of Mumbai Bharati Vidyapeeth College of Engineering, Sector-7, Belpada, Navi Mumbai, India

Abstract— Although personalized search has been proposed for many years and many personalization strategies have been investigated, it is still unclear whether personalization is consistently effective on different queries for different users, and under different search contexts. In this paper, we study this problem and provide some preliminary conclusions. We present a large-scale evaluation framework for personalized search based on query logs. Also, we reveal that personalized search has significant improvement over common web search on some queries but it has little effect on other queries (e.g., queries with small click entropy). It even harms search accuracy under some situations. We propose a PWS framework called UPS that can adaptively generalize profiles by queries while respecting user-specified privacy requirements. Our runtime generalization aims at striking a balance between two predictive metrics that evaluate the utility of personalization and the privacy risk of exposing the generalized profile.

Key words: Personalization, Privacy, User Profile, User Feedback

I. INTRODUCTION

As the amount of information on the web continuously grows, it has become increasingly difficult for web search engines to find information that satisfies users' individual needs. Personalized search is a promising way to improve search quality by customizing search results for people with different information goals. Many recent research efforts have focused on this area. Most of them could be categorized into two general approaches: Re-ranking query results returned by search engines locally using personal information; or sending personal information and queries together to the search engine [1]. A good personalization algorithm relies on rich user profiles and web corpus. However, as the web corpus is on the server, re-ranking on the client side is bandwidth intensive because it requires a large number of search results transmitted to the client before re-ranking. Alternatively, if the amount of information transmitted is limited through filtering on the server side, it pins high hope on the existence of desired information among filtered results, which is not always the case. Therefore, most of personalized search services online like Google Personalized Search and Yahoo! My Web adopt the second approach to tailor results on the server by analyzing collected personal information, e.g. personal interests, and search histories. One criticism of search engines is that when queries are issued, most return the same results to users. In fact, the vast majority of queries to search engines are short [2] and ambiguous, and different users may have completely different information needs and goals under the same query [2]. For example, a biologist may use query "mouse" to get information about rodents, while programmers may use the same query to find information about computer peripherals. When such a query is submitted to a search engine, it takes a moment for a user to choose which information he/she wishes to get. On the query "free mp3 download", the users' selections can also vary though almost all of them are finding some websites to download free mp3: one may select the website "www.yourmp3.net", while another may prefer the website "www.seekasong.com".

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 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. 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. There are two solutions to PWS, click-log-based methods and profile-based-methods. The former is bias to clicked URLs or pages in the particular user's history and can work only on repeated queries. In contrast to this, the latter improves the search experience with user-interest models [1]. These user interest models are generated from users' profiles. PWS has illustrated more effectiveness in improving the quality of web data search. For this, implicit user data has to be collected which can be collected from query history [2], browsing history, bookmarks [1], and click-through data [3]. This raises privacy issues due to the lack of protection of user's private data. This may raise panic among the users and can also smother the publisher's enthusiasm for offering such services. For protecting user privacy in profile-based PWS, developers have to consider two contradicting effects while performing the search process. They have to make an attempt to improve the search quality with the personalization utility and on the other hand they need to hide the privacy contents existing in the user profile for keeping the privacy risk under control [1]. People are willing to compromise their private data if this will help in an easy access to required information and an efficient search quality. A significant amount of gain can be obtained by personalizing users' information at the cost of a small information, a generalized profile.

People are not good at specifying detailed informational goals, so we use information about the searcher that we can glean in an automated manner to infer an implicit goal or intent. We explore the use of a very rich user profile, based both on search related information such as previously issued queries and previously visited Web pages, and on other information such as documents and email the user has read and created. Our research suggests that by treating the implicitly constructed user profile as a form of relevance feedback, we can obtain better performance than explicit relevance feedback and can improve on Web search.

Paper is organized in the following manner: Section two deals with literature survey, section 3 deals with the system architecture and section four provides the conclusion.

II. LITERATURE SURVEY

There are several prior attempts on personalizing web search. One approach is to ask users to specify general interests. The user interests are then used to filter search results by checking content similarity between returned web pages and user. Unfortunately, studies have also shown that the vast majority of users are reluctant to provide any explicit feedback on search results and their interests [4]. Many later works on personalized web search focused on how to automatically learn user preferences without any user efforts. User profiles are built in the forms of user interest categories or term lists/vectors. In [1], user profiles were represented by a hierarchical category tree based on ODP and corresponding keywords associated with each category. User profiles were automatically learned from search history. User preferences were built as vectors of distinct terms and constructed by accumulating past preferences, including both long-term and short-term preferences. In this paper, user profiles are represented as a hierarchical structure, and these profiles are also automatically learned from users' past clicked web pages. Nonetheless, this approach has privacy issues on exposing personal information to a public server. It usually requires users to grant the server full access to their personal and behaviour information on the Internet. Without the user's permission, gleaning such information would violate an individual's privacy.

In particular, Canada launched the Personal Information Protection and Electronic Document Act1 in 2001 to protect a wide spectrum of information, i.e., age, race, income, evaluations, and even intentions to acquire goods or services from being released to outside parties [3]. It is also evidenced by a recent survey conducted by Choicestream2 that the privacy fear continues to escalate although personalization remains something most consumers want. The number of consumers interested in personalization remains at a remarkably high 80%; however, only 32% of respondents were willing to share personal information in exchange for personalized experience, down from 41% in 2004. Recent coverage about identity thefts and online security breaches, i.e. AOL search query data scandal, even causes users to be more wary than ever on sharing their private information—even with established, trusted brands.

Many personalized web search strategies based on hyperlink structure of web have also been investigated. Personalized PageRank, which is a modification of the global PageRank algorithm, was first proposed for personalized web search. Multiple Personalized PageRank scores, one for each main topic of ODP, were used to enable "topic sensitive" web search. Most recently, researchers developed a method to automatically estimate user hidden interests based on Topic- Sensitive PageRank scores of the user's past clicked pages. In most of above personalized search strategies, only the information provided by user himself/herself is used to create user profiles. These are also some strategies which incorporate the preferences of a group of users to accomplish personalized search. In these approaches, the search histories of users who have similar interest with test user are used to refine the search. Collaborative filtering is a typical group-based personalization method and has been used in personalized search in [4].

A key factor for today's popular search engines is that they provide a user-friendly interface. The topics which are displayed on the web page related to a particular query are in the form of list of keywords entered by the user in the search bar, ranked according to their relevance with the original query. Ranking has become a central research problem for informational retrieval and Web data search, as it directly influences the relevance of the search results, the quality of a search system and users' search experience. Given a query, the deployed ranking function measures the relevance of each document to the query, sorts all the relevant documents and presents a list of top-ranked ones to the user. Despite of the simple interaction which proved to be successful, a list of keywords is not a good descriptor of the required information by the users. Users can not always formulate an efficient query to these search engines. One reason for this is the ambiguous data which is entered by the user. Often, users try different queries till get satisfied with the appropriate results. If users are familiar with the specific terminologies required, effective formulation can be achieved. But this may not be the case always. Users may have a little knowledge about what they are searching or even worse they do not what they are searching at all.

Privacy concerns are natural and important especially on the Internet. Some prior studies on Private Information Retrieval (PIR) [6], focuses on the problem of allowing the user to retrieve information while keeping the query private. Instead, this study targets preserving privacy of the user profile, while still benefiting from selective access to general information that the user agrees to release. To our knowledge, this problem has not been studied in the context of personalized search. One possible reason for this is that personal information, i.e. browsing history and emails, is mostly unstructured data, for which privacy is difficult to measure and quantify. These problems are addressed in the framework discussed in this paper i.e. User customizable Privacy preserving Search (UPS).

III. SYSTEM ARCHITECTURE

A. User Profile

Consistent with many previous works in personalized web services, each user profile in UPS adopts a hierarchical structure. Moreover, our profile is constructed based on the availability of a public accessible taxonomy, denoted as R, which satisfies the following assumption.

1) Assumption 1:

The repository R is a huge topic hierarchy covering the entire topic domain of human knowledge. That is, given any human recognizable topic t, a corresponding node (also referred to as t) can be found in R, with the subtree as the taxonomy accompanying t.

The repository is regarded as publicly available and can be used by anyone as the background knowledge. Such repositories do exist in the literature, for example, the ODP [4], Wikipedia, WordNet, and so on.

2) Assumption 2:

Given a taxonomy repository R, the repository support is provided by R itself for each leaf topic.

In fact, Assumption 2 can be relaxed if the support values are not available. In such case, it is still possible to "simulate" these repository supports with the topological structure of R. Based on the taxonomy repository, we define a probability model for the topic domain of the human knowledge. In the model, the repository R can be viewed as a hierarchical partitioning of the universe.

B. Query Representation

The document representation is important in determining both what terms are included and how often they occur. Using the full text of documents in the results set is a natural starting place. However, accessing the full text of each document takes considerable time. Thus, we also experimented with using only the title and the snippet of the document returned by the Web search engine. We note that because the Web search engine we used derived its snippets based on the query terms, the snippet is inherently query focused. Thus, for the query "cancer", if a document contained the following words,

The American Cancer Society is dedicated to eliminating cancer as a major health problem by preventing cancer, saving lives, and diminishing suffering through each word would affect the document score. To maintain a degree of emphasis on the query, we also tried selecting from the documents the subset of terms that were relevant to the query. This was done in a simple manner, by including the words that occurred near the query term.

C. Online Profiler

As illustrated in Fig. 1, UPS consists of a non-trusty search engine server and a number of clients. Each client (user) accessing the search service trusts no one but himself/herself. The key component for privacy protection is an online profiler implemented as a search proxy running on the client machine itself. The proxy maintains both the complete user profile, in a hierarchy of nodes with semantics, and the user-specified (customized) privacy requirements represented as a set of sensitive-nodes. The framework works in two phases, namely the offline and online phase, for each user. During the offline phase, a hierarchical user profile is constructed and customized with the user-specified privacy requirements. The online phase handles queries as follows:

- When a user issues a query qi on the client, the proxy generates a user profile in runtime in the light of query terms. The output of this step is a generalized user profile *Gi* satisfying the privacy requirements. The generalization process is guided by considering two conflicting metrics, namely the personalization utility and the privacy risk, both defined for user profiles.
- Subsequently, the query and the generalized user profile are sent together to the PWS server for personalized search.
- The search results are personalized with the profile and delivered back to the query proxy.
- Finally, the proxy either presents the raw results to the user, or re-ranks them with the complete user profile.

UPS is distinguished from conventional PWS in that it 1) provides runtime profiling, which in effect optimizes the personalization utility while respecting user's privacy requirements; 2) allows for customization of privacy needs; and 3) does not require iterative user interaction.

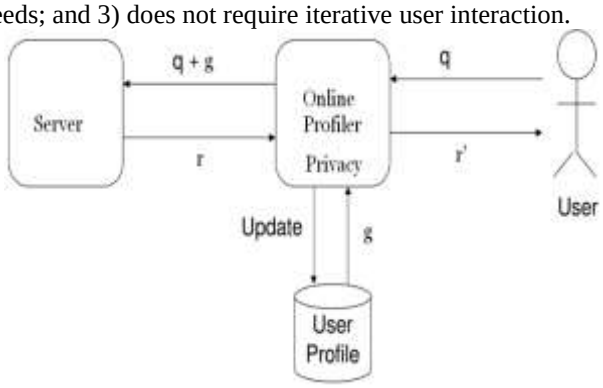

Fig. 1: Block Diagram of UPS

The profile-based personalization contributes a little and even reduces the quality of search when there is a large amount of distinct queries. This may expose the profile to the server and will risk the privacy of the user. There is a solution to this problem. The decision to personalize the users' profile or not can be made in the online phase. The idea behind this phase it very simple, if a query issued is a distinct query during generalization the complete runtime profiling will then be aborted. Then the query will be sent to the server without any user profile.

D. Attack Model

The user profile should be protected from antagonists which try to hinder the privacy and sensitive nodes defined by the user by a typical attack, namely eavesdropping. As shown in the Fig. 2 the eavesdropper intercepts successfully the communication happening between the server and the user by a measure, such as man-in-the-middle attack, invading the server. Accordingly, whenever the user issues any query q, the entire copy of q along with the runtime profile of the user will be seized the attacker.

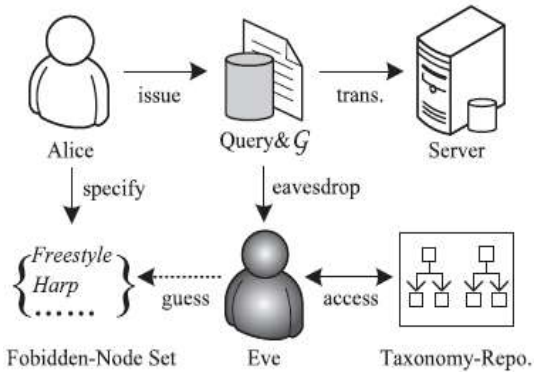

Fig. 2: Attack Model

The attacker will then try to recover the hidden segments defined as private by the user. Now, the adversary is considered to satisfy the following assumptions:

1) Knowledge Bound

The background knowledge of the adversary is limited to the entire information available on the web. Both the original

user profile and the privacy are defined within this information.

2) Session Bound

Previously captured information is not available for tracing the same victim. The eavesdropping will be started and ended within a single session.

To recover this problem, we built a generalized profile which consists of only the related or relevant information about the query. The unnecessary data is avoided from sending to the server. This also saves the computation time. Also, we provide a mechanism called online decision wherein the user has the authority to decide whether to personalize his/her query or not.

E. Attribute Partitioning

This algorithm partitions attributes so that highly correlated attributes are in the same column. This is good for both utility and privacy. In terms of data utility, grouping highly correlated attributes preserves the correlations among those attributes. In terms of privacy, the association of uncorrelated attributes presents higher identification risks than the association of highly correlated attributes because the associations of uncorrelated attribute values is much less frequent and thus more identifiable.

F. Tuple Partitioning

The algorithm maintains two data structures: 1) a queue of buckets Q and 2) a set of sliced buckets (SB). Initially, Q contains only one bucket which includes all tuples and SB is empty. For each iteration, the algorithm removes a bucket from Q and splits the bucket into two buckets. If the sliced table after the split satisfies l-diversity, then the algorithm puts the two buckets at the end of the queue Q Otherwise, we cannot split the bucket anymore and the algorithm puts the bucket into SB. When Q becomes empty, we have computed the sliced table. The set of sliced buckets is SB.

IV. CONCLUSION

In this paper, we try to investigate whether personalization is consistently effective under different situations. We develop an evaluation framework UPS to enhance personalized search. We find all proposed methods have significant improvements over common web search on queries with large click entropy. On the queries with small click entropy, the performance is enriched. These results tell us that personalized search has different effectiveness on different queries and thus not all queries should be handled in the same manner. Click entropy can be used as a simple measurement on whether the query should be personalized. They are straightforward and stable though they can work only on repeated queries.

In this paper, we investigated the feasibility of achieving a balance between users' privacy and search quality. First, an algorithm is provided to the user for collecting, summarizing, and organizing their personal information into a hierarchical user profile, where general terms are ranked to higher levels than specific terms. Through this profile, users control what portion of their private information is exposed to the server.

REFERENCES

[1] Zhicheng Dou, Ruihua Song, JiRong Wen, "A Large scale Evaluation and Analysis of personalized Search Strategies", WWW 2007, May 8–12, 2007, Banff, Alberta, Canada.
[2] Jaime Teevan, Susan T. Dumais, Eric Horvitz, "Personalizing Search via Automated Analysis of Interests and Activities", SIGIR '05, August 15–19, 2005, Salvador, Brazil.
[3] Mirco Speretta, Susan Gauch, "Personalizing Search Based on User Search Histories", SIGIR '04, January 1–2, 2004, California,USA.
[4] Kazunari Sugiyama, Kenji Hatano, Masatoshi Yoshikawa, "Adaptive Web Search Based on User Profile Constructed without Any Effort from Users", WWW2004, May 17–22, 2004, New York, New York, USA.
[5] Feng Qiu, Junghoo Cho, "Automatic Identification of User Interest For Personalized Search", WWW2006, May 22–26, 2006, Edinburgh, UK.
[6] Yabo Xu, Benyu Zhang, Zheng Chen, Ke Wang, "Privacy-Enhancing Personalized Web Search", WWW 2007, May 8–12, 2007, Banff, Alberta, Canada.
[7] Lidan Shou, He Bai, Ke Chen, and Gang Chen, "Supporting Privacy Protection in Personalized Web Search," Proc. IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING VOL:26 NO:2 YEAR 2014.

