

# ENHANCED PRIVACY PROTECTION IN PERSONALIZED WEB SEARCH

<sup>1</sup>. V MANIKYALA RAO, <sup>2</sup>. P LAKSHMAN SWAMY

<sup>1</sup>. PG Scholar, Dept of CSE, Sri Sunflower College of Engineering And Technology, Lankapalli, AP

<sup>2</sup>. ASSOCIATE PROFESSOR, Dept of CSE, Sri Sunflower College of Engineering And Technology, Lankapalli, AP

## **ABSTRACT**

Personalized web search (PWS) has demonstrated its effectiveness in improving the quality of various search services on the Internet. However, evidences show that users' reluctance to disclose their private information during search has become a major barrier for the wide proliferation of PWS. We study privacy protection in PWS applications that model user preferences as hierarchical user profiles. 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. We present two greedy algorithms, namely GreedyDP and GreedyIL, for runtime generalization. We also provide an online prediction mechanism for deciding whether personalizing a query is beneficial. Extensive experiments demonstrate the effectiveness of our framework. The experimental results also reveal that GreedyIL significantly outperforms GreedyDP in terms of efficiency.

## **INTRODUCTION**

The web search engine has long become the most important portal for ordinary people looking for useful information on the web. However, users might experience failure when search engines return irrelevant results that do not meet their real intentions. Such

irrelevance is largely due to the enormous variety of users' contexts and backgrounds, as well as the ambiguity of texts. Personalized web search (PWS) is a general category of search techniques aiming at providing better search results, which are tailored for individual user needs. As the expense, user information has to be

collected and analyzed to figure out the user intention behind the issued query.

The solutions to PWS can generally be categorized into two types, namely click-log-based methods and profile-based ones. The click-log based methods are straightforward—they simply impose bias to clicked pages in the user's query history. Although this strategy has been demonstrated to perform consistently and considerably well, it can only work on repeated queries from the same user, which is a strong limitation confining its applicability. In contrast, profile-based methods improve the search experience with complicated user-interest models generated from user profiling techniques. Profile-based methods can be potentially effective for almost all sorts of queries, but are reported to be unstable under some circumstances.

Although there are pros and cons for both types of PWS techniques, the profile-based PWS has demonstrated more effectiveness in improving the quality of web search recently, with increasing usage of personal and behavior information to profile its users, which is usually gathered implicitly from query history, browsing history, click-through data, bookmarks, user documents, and so forth. Unfortunately, such implicitly

collected personal data can easily reveal a gamut of user's private life. Privacy issues arising from the lack of protection for such data, for instance the AOL query logs scandal, not only raise panic among individual users, but also dampen the data-publisher's enthusiasm in offering personalized service. In fact, privacy concerns have become the major barrier for wide proliferation of PWS services.

### 1.1 Motivations

To protect user privacy in profile-based PWS, researchers have to consider two contradicting effects during the search process. On the one hand, they attempt to improve the search quality with the personalization utility of the user profile. On the other hand, they need to hide the privacy contents existing in the user profile to place the privacy risk under control. A few previous studies suggest that people are willing to compromise privacy if the personalization by supplying user profile to the search engine yields better search quality. In an ideal case, significant gain can be obtained by personalization at the expense of only a small (and less-sensitive) portion of the user profile, namely a generalized profile. Thus, user privacy can be protected without compromising the

personalized search quality. In general, there is a tradeoff between the search quality and the level of privacy protection achieved from generalization.

Unfortunately, the previous works of privacy preserving PWS are far from optimal. The problems with the existing methods are explained in the following observations:

1. The existing profile-based PWS do not support runtime profiling. A user profile is typically generalized for only once offline, and used to personalize all queries from a same user indiscriminately. Such “one profile fits all” strategy certainly has drawbacks given the variety of queries. One evidence reported in this system is that profile-based personalization may not even help to improve the search quality for some ad hoc queries, though exposing user profile to a server has put the user’s privacy at risk. A better approach is to make an online decision on

- a. whether to personalize the query (by exposing the profile) and
- b. what to expose in the user profile at runtime.

To the best of our knowledge, no previous work has supported such feature.

2. The existing methods do not take into account the customization of privacy requirements. This probably makes some user privacy to be overprotected while others insufficiently protected. For example, in this system, all the sensitive topics are detected using an absolute metric called surprisal based on the information theory, assuming that the interests with less user document support are more sensitive. However, this assumption can be doubted with a simple counterexample: If a user has a large number of documents about “sex,” the surprisal of this topic may lead to a conclusion that “sex” is very general and not sensitive, despite the truth which is opposite. Unfortunately, few prior work can effectively address individual privacy needs during the generalization.

3. Many personalization techniques require iterative user interactions when creating personalized search results. They usually refine the search results with some metrics which require multiple user interactions, such as rank scoring, average rank, and so on. This paradigm is, however, infeasible for runtime profiling, as it will not only pose too much risk of privacy breach, but also demand prohibitive processing time for profiling. Thus, we need predictive metrics

to measure the search quality and breach risk after personalization, without incurring iterative user interaction.

### 1.2 Contributions

The above problems are addressed in our UPS (literally for User customizable Privacy-preserving Search) framework. The framework assumes that the queries do not contain any sensitive information, and aims at protecting the privacy in individual user profiles while retaining their usefulness for PWS.

As illustrated in this system, UPS consists of a nontrusty 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:

1. When a user issues a query  $q_i$  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  $G_i$  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.
2. Subsequently, the query and the generalized user profile are sent together to the PWS server for personalized search.
3. The search results are personalized with the profile and delivered back to the query proxy.
4. Finally, the proxy either presents the raw results to the user, or reranks 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. Our main contributions are summarized as following:

We propose a privacy-preserving personalized web search framework UPS, which can generalize profiles for each query according to user-specified privacy requirements. Relying on the definition of two conflicting metrics, namely personalization utility and privacy risk, for hierarchical user profile, we formulate the problem of privacy-preserving personalized search as  $\epsilon$ -Risk Profile Generalization, with its NP-hardness proved. We develop two simple but effective generalization algorithms, GreedyDP and GreedyIL, to support runtime profiling. While the former tries to maximize the discriminating power (DP), the latter attempts to minimize the information loss (IL). By exploiting a number of heuristics, GreedyIL outperforms GreedyDP significantly. We provide an inexpensive mechanism for the client to decide whether to personalize a query in UPS. This decision can be made before each runtime profiling to enhance the stability of the search results while avoid the unnecessary exposure of the profile.

. Our extensive experiments demonstrate the efficiency and effectiveness of our UPS framework.

## **SYSTEM ANALYSIS**

### **EXISTING SYSTEM:**

The solutions to PWS can generally be categorized into two types, namely click-log-based methods and profile-based ones. The click-log based methods are straightforward—they simply impose bias to clicked pages in the user's query history. Although this strategy has been demonstrated to perform consistently and considerably well [1], it can only work on repeated queries from the same user, which is a strong limitation confining its applicability. In contrast, profile-based methods improve the search experience with complicated user-interest models generated from user profiling techniques. Profile-based methods can be potentially effective for almost all sorts of queries, but are reported to be unstable under some circumstances.

### **DISADVANTAGES OF EXISTING SYSTEM:**

The existing profile-based PWS do not support runtime profiling. The existing methods do not take into account the customization of privacy requirements.

Many personalization techniques require iterative user interactions when creating personalized search results. Generally there are two classes of privacy protection problems for PWS. One class includes those treat privacy as the identification of an individual, as described. The other includes those consider the sensitivity of the data, particularly the user profiles, exposed to the PWS server

### **PROPOSED SYSTEM:**

We propose a privacy-preserving personalized web search framework UPS, which can generalize profiles for each query according to user-specified privacy requirements. Relying on the definition of two conflicting metrics, namely personalization utility and privacy risk, for hierarchical user profile, we formulate the problem of privacy-preserving personalized search as  $\#$ -Risk Profile Generalization, with its N P-hardness proved. We develop two simple but effective generalization algorithms, GreedyDP and GreedyIL, to support runtime profiling. While the former tries to maximize the discriminating power (DP), the latter attempts to minimize the information loss (IL). By exploiting a

number of heuristics, GreedyIL out performs GreedyDP significantly.

We provide an inexpensive mechanism for the client to decide whether to personalize a query in UPS. This decision can be made before each runtime profiling to enhance the stability of the search results while avoid the unnecessary exposure of the profile. Our extensive experiments demonstrate the efficiency and effectiveness of our UPS framework.

### **ADVANTAGES OF PROPOSED SYSTEM:**

Increasing usage of personal and behaviour information to profile its users, which is usually gathered implicitly from query history, browsing history, click-through data bookmarks, user documents, and so forth. The framework allowed users to specify customized privacy requirements via the hierarchical profiles. In addition, UPS also performed online generalization on user profiles to protect the personal privacy without compromising the search quality.

**INPUT DESIGN** The input design is the link between the information system and the user. It comprises the developing



specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

- What data should be given as input?
- How the data should be arranged or coded?
- The dialog to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error occur.

## OBJECTIVES

1. Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input

process and show the correct direction to the management for getting correct information from the computerized system.

2. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.

3. When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow

## 2.2 OUTPUT DESIGN

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and

intelligent output design improves the system's relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.

2. Select methods for presenting information.

3. Create document, report, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

- ❖ Convey information about past activities, current status or projections of the
- ❖ Future.
- ❖ Signal important events, opportunities, problems, or warnings.
- ❖ Trigger an action.
- Confirm an action.

## CONCLUSION

This system presented a client-side privacy protection framework called UPS for personalized web search. UPS could potentially be adopted by any PWS that captures user profiles in a hierarchical taxonomy. The framework allowed users to specify customized privacy requirements via the hierarchical profiles. In addition, UPS also performed online generalization on user profiles to protect the personal privacy without compromising the search quality. We proposed two greedy algorithms, namely GreedyDP and GreedyIL, for the online generalization. Our experimental results revealed that UPS could achieve quality search results while preserving user's customized privacy requirements. The results also confirmed the effectiveness and efficiency of our solution.

For future work, we will try to resist adversaries with broader background knowledge, such as richer relationship among topics (e.g., exclusiveness, sequentiality, and so on), or capability to capture a series of queries (relaxing the second constraint of the adversary in Section 3.3) from the victim. We will also seek more sophisticated method to build the user



profile, and better metrics to predict the performance (especially the utility) of UPS.

### **REFERENCES**

[1] Z.Dou, R. Song, and J.-R. Wen, "A Large-Scale Evaluation and Analysis of Personalized Search Strategies," Proc. Int'l Conf. World Wide Web (WWW), pp. 581-590, 2007.

[2] J. Teevan, S.T. Dumais, and E. Horvitz, "Personalizing Search via Automated Analysis of Interests and Activities," Proc. 28th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR), pp. 449-456, 2005.

[3] M.Spertta and S. Gach, "Personalizing Search Based on User Search Histories,"

Proc. IEEE/WIC/ACM Int'l Conf. Web Intelligence WI), 2005.

[4] B. Tan, X. Shen, and C. Zhai, "Mining Long-Term Search History to Improve Search Accuracy," Proc. ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD), 2006.

