Journal of Science and Technology (JST)
Volume 3, Issue 1, Jan - February 2018, PP 17-21
www.jst.org.in *ISSN: 2456 - 5660*

Supporting Privacy Protection in Personalized Web Search

[1]Brahmaji Katragadda, [2]Sk.Meera Sharife

[1](M. Tech, Dept. of CSE, GIET College, Rajahmundry)
[2](Associate Professor, Dept. of CSE, GIET College, Rajahmundry)

Abstract -*The potency of Personalized web search (PWS) inenhancing the quality of diverse search services on the Internetis authenticated. Nevertheless, user's disinclination to unfold their private information in the course of their search has created a vitalstop for the proliferation of PWS. We aspire to propose a PWS framework called UPS. while valuing user specified privacy requirements, with the help of queries, this can adaptively generalize profiles.This technique aims at maintaining equilibrium between two predictive metrics that gauges the utility of personalization and the privacy risk of uncovering the generalized profile. GreedyDP and GreedyIL are the two greedy algorithms for runtime generalization are proposed here. Additionally, weimpart an online prediction mechanism for deciding whether personalizing a query is serviceable. Extensive experiments demonstrate the effectiveness of our framework. The experimental results also reveal that GreedyIL outstandingly surpasses GreedyDP in terms of efficiency.*

Keywords - *Privacy protection, personalized web search, utility, risk, profile*

I. INTRODUCTION

The extended tool for accessing the data anywhere in the world is at arm's length as the consequence of web search engine. Still, search engines fail to retrieve relevant results which outbound the user's expectations. This is due to wide reaching variety of users' contexts, backgrounds and additionally the text ambiguity. Personalized web search (PWS) is a general category of search techniques directed to provide appropriate search results, which are fashioned or users. As the expense, user information has to be hoarded and analyzed base on the issued query.PWS can be categorized into two types, namely click-log-based and profile-based methods. The click-log based methods are straightforward—they use the history of user's query and simply impose bias to clicked pages. This can only work on repeated queries from the invariable user, which limits its usefulness. But, profile-based methods fostered from archetypes generated from user profiling techniques revamp the search experience within tricated user-interest. Profile-based methods are effective for almost all sorts of queries, but under some circumstances they are suspected to be unstable.

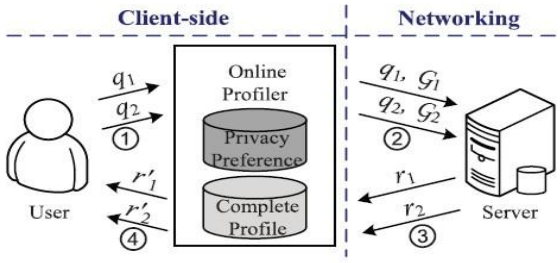
2. The prevailing methods do not take into account the customization of privacy requirements. This seems to makes some user privacy to be overprotected while others leaving under protected. Consider the example as, in all the subtle topics are uncovered using an absolute metric called surprisal based on the information theory, presuming that the interests with less user document support are fragile. However, this assumption can be suspected with a simple counterexample: If a user has a large number of documents about "sex," the surprisal of this topic may conclude that "sex" is very general and not sensitive, despite the fact which is opposite. Luckily, few prior work scan address privacy needs of the user during the generalization.

3. When creating personalized search results, many personalization techniques require iterative user interactions. They typically filter the search results with some metrics which entails numerous user interactions, such as rank scoring, average rank, and so on.

This archetype, however is impracticable for runtime profiling, as it opens up roads to much risk of privacy breach, but also demand prohibitive processing time for profiling. Thus, we require predictive metrics to measure the search quality and breach risk after personalization, without incurring iterative user interaction.

1.1 Contributions

Our UPS (literally for User Customizable Privacy-preserving Search) frame work addressed the above problems. The framework assumes that the queries do not hold any fragile information, and aims at insulating the privacy in individual user profiles while preserving their usefulness for PWS.



1. System architecture of UPS.

As illustrated in Fig. 1, UPS consists of a number of clients and a non trusty search engine server. Each client (user) retrieving the search service trusts no one except himself/ herself. The essential component for privacy protection is an online profiler enact as a search proxy running on the client machine itself. The proxy retains 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 every individual user. During the offline phase, a hierarchical user profile is created and customized with the user-specified privacy requirements. The online phase direct queries as follows:

1. When a query qi is issued by a user on the client, a user profile is generated by the proxy in the light of query terms in runtime. A generalized user profile Gi is the output of this step fulfills 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. Successively, for personalized search the query and the generalized user profile are sent together to the PWS server.
3. Corresponding to the profile the search results are personalized and delivered back to the query proxy.
4. Eventually, the proxy either presents the user with the raw results, or reranks them with the complete user profile.

UPS is distinguished from conventional PWS in that it

1) renders runtime profiling, which in turn optimizes the personalization utility while respecting user's privacy requirements;
2) allowing privacy needs for customization; and
3) iterative user interaction is not required.

The following summarizes our main contributions:

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 NP-hardness proved. We develop two effective generalization algorithms which are simple, Greedy IL and Greedy DP, to support runtime profiling. While the former attempts to minimize the information loss (IL), the latter tries to maximize the discriminating power (DP). After exploiting a number of heuristics, Greedy IL out performs Greedy DP significantly. An inexpensive mechanism is provided by us for the client to decide whether to personalize a query in UPS. Before each runtime profiling this decision is made to enhance the stability of the search results while avoid the unnecessary exposure of the profile. The efficiency and effectiveness of our UPS framework are demonstrated by our experiments.

II. RELATED WORKS

In this section, we focus on the literature of profile-based personalization and privacy protection in PWS system.

Profile-Based Personalization

Previous works on profile-based PWS primarily focus on enhancing the search utility. These works on tailoring the search results by referring to, often implicitly, a user profile that discloses an individual information goal. In the remainder of this section, we review the previous solutions to PWS on two facets, namely the representation of profiles, and gauging the effectiveness of personalization.

Privacy Protection in PWS System

Predominantly there are two classes of privacy protection problems for PWS. One class includes those treat privacy as the recognition of an individual, as described in. The other includes those consider the vulnerability of the data, remarkably the user profiles, uncovering to the PWS server. Typical works in the literature of

protecting user identifications (class one) strives to decode the privacy problem on disparate volumes, including the pseudo and the group identity, no identity, and no personal information. These works presume the existence of a reliable third-party anonymizer, which, over the internet is not readily handy. One main restriction is that it builds the user profile as a finite set of attributes, and the probabilistic model is trained through frequent predefined queries. These assumptions in the context of PWS are not practical. A more important property that differentiate our work from [1] is that we present personalized privacy protection in PWS. The theory of personalized privacy protection is first pioneered by Xiao and Tao [2] in Privacy-Preserving Data Publishing (PPDP). Any individual can define the degree of privacy protection for her/his sensitive values by defining "guarding nodes" in the taxonomy of the sensitive attribute. Motivated by this, we allow users in their hierarchical user profilesto customize privacy needs.

Ultimately, we present the attack model and formulate the problem of privacy preserving profile generalization. For ease of understanding the presentation, In the below Table 1 all the symbols used in this paper are summarized.

TABLE 1
Symbols and Descriptions

Symbol	Description
$\|\mathcal{T}\|$	The count of nodes of the tree $\mathcal{T}$
$t \in \mathcal{T} / N \subset \mathcal{T}$	t is a node (N is a node set) in the tree $\mathcal{T}$
$subtr(t, \mathcal{T})$	The subtree rooted on t within the tree $\mathcal{T}$
$rsbtr(N, \mathcal{T})$	The rooted subtree of $\mathcal{T}$ by removing the set N
$trie(N)$	The topic-path prefix tree built with the set N
$root(\mathcal{T})$	The root of the tree $\mathcal{T}$
$par(t, \mathcal{T})$	The parent of t in the tree $\mathcal{T}$
$lca(N, \mathcal{T})$	The least common ancestor of the set N in $\mathcal{T}$
$C(t, \mathcal{T})$	The children of t within the tree $\mathcal{T}$

III. PROBLEM DEFINITION

User Profile

In consistent with various earlier works in personalized web services, UPS adopts a hierarchical structure for every user profile. Furthermore, based on the availability of a public accessible tax onomyour profile is constructed, denoted as R, which satisfies the below assumption.

*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 sub trees ubtrðt; RÞ as the taxonomy accompanying. The repository is regarded as open source and can be used by anyone as the background knowledge. Such repositories do exist in the literature, for example, the OD Pand soon. Moreover, each topic is associated with are pository support, denoted by which quantifies how often the respective topic influences the human knowledge. If we examine every topic as a result of a random walk from its parent topic in R, we get the following recursive equation:

$$sup_{\mathcal{R}}(t) = \sum_{t' \in C(t,\mathcal{R})|} sup_{\mathcal{R}}(t'). \qquad (1)$$

Equation (1) can be utilized to evaluate the repository support of all topics in R, relying on the assumption that he support values of all leaf topics in Rare accessible.

*Assumption 2.*Given a taxonomy repository R, the repository support for each leaf topic is provided by R itself. In fact, If the support values are not available assumption 2 can be relaxed. Under such circumstances, it is still possible with the topological structure of Rto "simulate" these repository supports. That is, calculated as the count of leaves in R.

For the topic domain of the human knowledge we define a probability model based on the taxonomy repository. In this model, the repository R can be observed as ahierarchical partitioning of the universe (represented by the root topic) and every topic t 2 R stands for a random event. The conditional probability Prðt j sÞ (s is an ancestor of t) is defined as the proportion of repository support:

$$Pr(t \mid s) = \frac{sup_{\mathcal{R}}(t)}{sup_{\mathcal{R}}(s)}, \quad t \in subtr(s, \mathcal{R}). \qquad (2)$$

$$Pr(t) = Pr(t \mid root(\mathcal{R})), \qquad (3)$$

Customized Privacy Requirements

Customized privacy requirements can be detailed in the user profile with a number of sensitive-nodes (topics), upon disclosure (to the server) introduces privacy risk to the user.

Attack Model

Our work targeted to provide protection against a particular model of privacy attack, called as eavesdropping. As shown in Fig. 3, the eavesdropper attempts to corrupt Alice's privacy, Eve successfully intercepts the communication between Alice and the PWS-server via some measures, such as man-in-the middle attack, pervade the server, and so on.

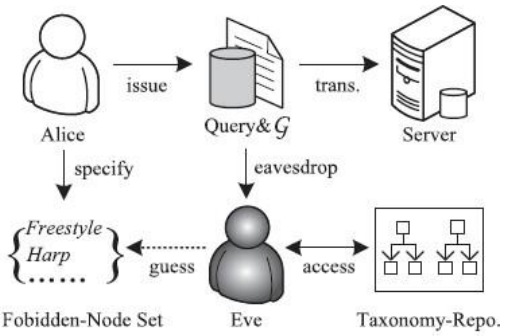


Fig. 3. Attack model of personalized web search.

As a result, Eve will hook up the entire copy of q together with a runtime profile G, whenever query q is delivered by Alice. Based on G, Eve will attempt to touch the sensitive nodes of Alice by recuperating the segments masked from the original H and computing a confidence for each recovered topic, depending on the background knowledge in the publicly available taxonomy repository R.

4. Effective Analysis of Personalization using our UPS framework, we assess and evaluate the real search quality on commercial search engines. A juxtaposing the personalization results of ODP and Yahoo unveils that, although the original ODP Rank (AP ¼ 37:3%) is lower than the original Yahoo- Rank (AP ¼ 46:7%), personalization on ODP will trigger improved ranking than that on Yahoo. The search results is reranked with the generalized profile output by Greedy IL over 50 target users. The final search quality is analyzed using the Average Precision of the click records of the users, which is defined as

$$\mathcal{AP} = \sum_{i=1}^{n} \frac{i}{l_i.rank}/n,$$

where li is the ith relevant link identified for a query, and nis the number of relevant links.

IV. CONCLUSION

This paper 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 secure the personal privacy with the improved search quality. Forthe online generalization we proposed two greedy algorithms, namely Greedy DP and Greedy IL,. 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] Y. Xu, K. Wang, B. Zhang, and Z. Chen, "Privacy-EnhancingPersonalized Web Search," Proc. 16th Int'l Conf. World Wide Web(WWW), pp. 591-600, 2007.

[2] X. Xiao and Y. Tao, "Personalized PrivacyPreservation," Proc.ACM SIGMOD Int'l Conf. Management of Data (SIGMOD), 2006.

[3] X. Shen, B. Tan, and C. Zhai, "Implicit User Modeling forPersonalized Search," Proc. 14th ACM Int'l Conf. Information and Knowledge Management (CIKM), 2005.

[4] X. Shen, B. Tan, and C. Zhai, "Context-Sensitive InformationRetrieval Using Implicit Feedback," Proc. 28th Ann. Int'l ACMSIGIR Conf. Research and Development Information Retrieval (SIGIR),2005.
[5] K. Ramanathan, J. Giraudi, and A. Gupta, "Creating HierarchicalUser Profiles Using Wikipedia," HP Labs, 2008.