International Journal of Engineering Research and General Science Volume 3, Issue 2, Part 2, March-April, 2015
ISSN 2091-2730

Privacy Protection in Personalized Web Search by hiding sensitive nodes of hierarchical user profile using GreedyDP and GreedyIL

Uday Dattatraya More
(*Student, Dr .D. Y. Patil Insti. Of Engg. & Tech., India*)

Kavita Tushar Nanekar
(*Student, Dr .D. Y. Patil Insti. Of Engg. & Tech., India*)
(pokharkarkavita@gmail.com)

Sana Sahar Murtuza Sothe
(*Student, Dr .D. Y. Patil Insti. Of Engg. & Tech., India)*

Sharmila A.Chopade
(*Faculty, Dr .D. Y. Patil Insti. Of Engg. & Tech., India)*
(ssothe@rediffmail.com)

Abstract-Personalized web search (PWS) has provided its effectiveness in improving the quality of various search services on the Internet. Personalized search is a promising way to improve the accuracy of web search, and has been attracting much attention now days. But effective, personalized search requires aggregating and collecting user information, which cause privacy infringement for many users; these infringements have become one of the main obstacles to deploying personalized search applications, and great challenge of how to do privacy preserving personalization. We study privacy protection in PWS applications that model user preference as hierarchical user profiles. We propose a PWS framework called UPS (User customizable Privacy-preserving Search) that can adaptively generalize profiles by queries while respecting user specified privacy requirements. Our runtime generalization has aims of keeping a balance between two predictive metrics that evaluate the utility of personalization and the privacy risk of exposing the user generalized profile.

Keyword- Privacy, Taxonomy, Web search, Servers, Sensitivity, profile, Privacy protection, personalized web search, utility, risk

1 Introduction:

Web search engines have made enormous contributions to the web and society. They make finding information on the web quick and easy. However, they are far from optimal. For a given query, a personalized Web search can provide different results for different users or organize results differently to each user, based upon their interests and information needs.Personalized web search is differs from generic web search because it returns identical research results to all users for identical queries, independent of varied user interests and information needs. A major deficiency of generic search engines is that they follow the ‘‘one size fits all’’ model and are not adaptable to individual users. This is typically shown in cases such as these:

1) Different users have different backgrounds and interests. They may have completely different information needs and goals when providing exactly the same query. For example, a biologist may issue ‘‘mouse’’ to get information about rodents, while programmers may use the same query to find information about computer peripherals. When such a query is issued, generic search engines will return a list of documents on different topics. It takes time for a user to choose which information he/she really wants, and this makes the user feel less satisfied. Queries like ‘‘mouse’’ are usually called ambiguous queries. Statistics has shown that the vast majority of queries are short and ambiguous. Generic web search usually fails to provide optimal results for ambiguous queries.
2) 2. Users are not static. User information needs may change over time. Indeed, users will have different needs at different times based on current circumstances. For example, a user may use ‘‘mouse’’ to find information about rodents when the user is viewing television news about a plague, but would want to find information about computer mouse products when purchasing a new computer.Generic search engines are unable to distinguish between such cases. Personalized web search is considered a promising solution to these problems, so it can provide different search results based upon the information as per user need. It exploits user information and search context in learning to which sense a query refers. Consider the query ‘‘mouse’’ mentioned above: Personalized web search can disambiguate the query by gathering the following user information:
   1. The user is a computer programmer not a biologist.
   2. The user has just input a query ‘‘keyboard,’’ but not ‘‘biology’’ or ‘‘genome.’’ Before entering this query, the user had just viewed a web page with many words related to computer mouse, such as ‘‘computing,’’ ‘‘input device,’’ and ‘‘keyboard.’’

Such irrelevance is largely due to the enormous variety of users contexts and backgrounds as well as the ambiguity of text. Personalized web search (PWS) is a general category of search techniques aiming at providing better search results, which are adjust for individual user needs. As the expenditure, 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 divided into two types

1. Click -Log –Based :-

The click-log based methods are clear-cut and simple, they simply impose bias to clicked pages in the user's query history, although this method has been demonstrated to perform consistently and remarkably well, it can only work on repeated queries from the same user, which is a strong limitation enclose its application.

2. Profile Based :-

To provide personalized search results to users, personalized web search maintains a user profile for each individual. A user profile stores approximations of user tastes, interests. It is generated and updated by exploiting user-related information. Such information may include:

a) Demographic and geographical information, including age, gender, education, language, country, address, interest areas, and other information.
b) Search history, including previous queries and clicked documents. User browsing behavior when viewing a page, such as dwelling time, mouse click, mouse movement, scrolling, printing, and bookmarking, is another important element of user interest.
c) Other user documents, such as bookmarks, favorite web sites, visited pages, and emails. The external user data stored in a user client is useful to personalize individual search results.

User information can be specified by the user (explicitly collecting) or can be automatically learnt from a user's historical activities (implicitly collecting). As the vast majority of users are reluctant to provide any explicit feedback on search results and users interests, many works on personalized web search focus on how to automatically learn user preferences without involving any direct user efforts. Collected user information is processed and organized as a user profile in a hierarchical structure, depending on the need of personalization algorithm. This can be completed by creating vectors of URLs/domains, keywords, topic categories or the like.

Although there are pros and cons for both types of PWS techniques, the profile-based PWS has indicate, more effectiveness in improving the quality of web search freshly, with increasing usage of personal and behavior information to profile its users. The users' reluctance to disclose their private information during search has become a major barrier for the wide proliferation of PWS. Privacy issues are rising from the lack of protection for such data. In fact, privacy concerns have become the major barrier for wide proliferation of PWS services.

Server side and client side implementation:

Personalized web search can be implemented on either server side (in the search engine) or client side (in the user's computer or a personalization agent). For server-side personalization, user profiles are construct, updated, and stored on the search engine side. User information is directly include into the ranking process, or is used to help process earliest search results. The advantage of this architecture is that the search engine can use all of its resources, for example link structure of the complete web, in its personalization algorithm. Also, the personalization algorithm can be easily accommodate without any client efforts. This architecture is accepted by some general search engines such as Google Personalized Search. The disadvantage of this architecture is that it brings high storage and computation costs when millions of users are using the search engine, and it also enhance privacy concerns when information about users is stored on the server.

For client-side personalization, users information is collected and stored on the client side (in the user's computer or a personalization agent), usually by installing a client software or plug-in on a user's computer. In client side, not only the user's search behavior but also his contextual activities (e.g., web pages viewed before) and personal information (e.g., emails, documents, and bookmarks) could be incorporated into the user profile. This allows the construction of a much richer user model for personalization. Privacy concerns are also reduced since the user profile is strictly stored and used on the client side. Another benefit is that the raised in computation and storage for personalization can be distributed among the clients. A main drawback of personalization on the client side is that the personalization algorithm cannot use some knowledge that is only available on the server side (e.g., Page Rank score of a result document). Furthermore, due to the limits of network bandwidth, the client can usually only process limited top results.

Challenges of Personalized Search

Despite the attractiveness of personalized search, there is no large-scale use of personalized search services currently. Personalized web search faces several challenges that retard its real-world large-scale applications:

1. Privacy is an issue. Personalized web search, especially server-side implement, requires collecting and aggregating a lot of user information including query and click through history. A user profile can reveal a large amount of private user information, such as hobbies, vocation, income level, and political inclination, which is clearly a serious concern for users. This could make many people nervous and feel afraid to use personalized search engines. A personalized web search will be not well received until it handles the privacy problem well.
2. It is really hard to infer user information needs accurately. Users are not static. They may randomly search for something which they are not interested in. They even search for other people sometimes. User search histories inevitably contain noise that is irrelevant or even harmful to current search. This may make personalization strategies unstable.
3. Queries should not be handled in the same manner with regard to personalization. Personalized search may have little effect on some queries. Some work investigates whether current web search ranking might be sufficient for clear/unambiguous queries and thus personalization is unnecessary.

2 Literature survey review:

1) A LargeScale Evaluation and Analysis of Personalized Search Strategies
Author: Z.Dou, R.Song, and J.-R. Wen,2007 Proc. Int'l Conf.
Method: A large scale evaluation framework for personalized search based on query logs,and then evaluate five personalized search strategies (incuding two click log based and profile based)using 12-days MSR query log.
Advantage: Search accuracy is evaluated by real user clicks recorded in query logs automatically.
Disadvantage: Personalization may lack effectiveness on some querry

2) Implicit User Modeling for Personalized Search
Author: sX. Shen,B.Tan, and C.Zhai, pro 14th ACM Int'l Conf. Information and Knowledge Management (CIKM), 2005
Method:Here present a decision theoretic framework and develop techniques for implicit user modeling in information retrieval. They develop an intelligent clientside web search agent (UCAIR) that can perform eager implicit feedback.
Advantage: Search agent can improve search accuracy over the popular Google search engine.
Disadvantage: They generally lack user modeling and are not adaptive to individual users.

3) Personalizing Adaptive Web Search Based on User Profile Constructed without any effort from Users.
Author: K. Sugiyama, K. Hatano, and M. Yoshikawa Proc. 13th Int'l Conf., 2004.
Method: Propose several approaches to adapting search results according to each user's need for relevant information without any user effort, and then verify the effectiveness of our proposed approaches.
Advantage: User's preferences can be achieved by user profile based on modified collaborative filtering with detailed analysis of user's browsing history in one day.
Disadvantage: Each user needs different information for his/her query. Therefore, the search results should be adapted to users with different information needs.

4) Mining Long-Term Search History to Improve Search Accuracy
Author: B.Tan,X.Shen, and C.Zhai,Proc. ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (KDD), 2006
Method: Statistical language modeling based methods to mine contextual information from long term search history.
Advantage: Exploit it for a more accurate estimate of query language model.
Disadvantages: The web search engines, suffer from the problem of documents to return is "one size fits all" the decision of which documents to return is based on query, without consideration of a particular.

3 System Description:

UPS 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. UPS Framework which generalized profile s for each query according to user specified privacy requirements. The problems of privacy preserving personalized search as Risk Profile Generalization, with its NP-Hardness proved. A Trade o_ between search quality and level of privacy protection achieved from generalization. Generalization algorithms are used namely GreedyDP and GreedyIL to find out an utilization of user search and improving performance.

1. User fire a query 'q' through a proxy refer as an on-line profiler to the server.
2. Then generalized profile is created by a proxy and both generalized profile and query are passed to the server.

3. Server gives response 'r' back to the proxy, then it decides either to re-ranked the search or provide as it is result to client as per the query.

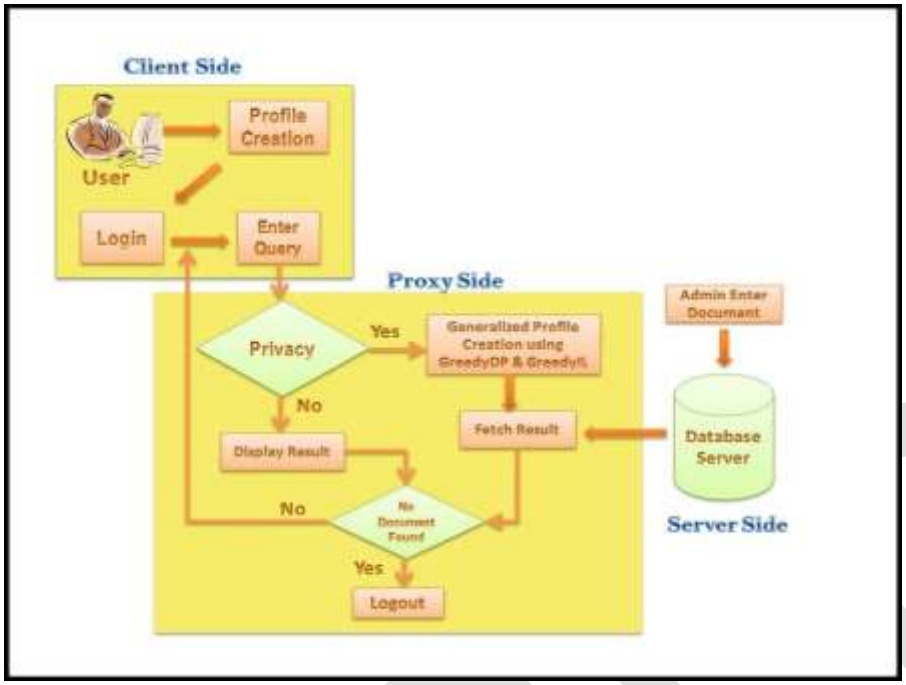

Fig:System Architecture

Advantages

1. PWS System enhances the stability of the search quality.
2. System avoids the unnecessary exposure of the user profile.
3. PWS System provides runtime profiling, which in effect optimizes the personalization utility while respecting user's privacy requirements.
4. System allows for customization of privacy needs.
5. Does not require iterative user interaction.
6. System is Client side completely.

Limitation

1. System is depends upon Proxy Server.
2. Proxy Server Failure fails the whole system.
3. System gives the results in "Text " format.
4. Only one user Login at a time to System.

3.1 Modules :

1. User Registration And Login Module :

This module accepts username , id and password and authenticates the user after user registration .The basic task of this module is to login system and obtained services from Server.

2. Profile Construction Module :

This module consist of a pro_le construction followed by:
(a) First assume that the user's preferences are represented in a set of plain text documents, denoted by D.
(b) Detect the respective topic in R for every document d is a element of D and split the query by using "@" sign.

3. Privacy Customization Module :

This module includes the Customization of user by users pro_le by calculating the cost of each node from the Document D. And Specifying the sensitive node and Non sensitive node. Provides the Privacy by hiding sensitive node.

4.Query Topic Mapping Module :

This module consist of query topic mapping which gives the seed profile in terms of result. Mapping is takes place in between Query and Taxonomay profile 'T'.All non-sensitive nodes set to 0 ,and removes the sensitive node from profile by applying Prune-leaf technique.

5.Profile Generalization Module :

This Module involves the Generalization of profile using seed profile and query by applying fusion of GreedyDP,GreedyIL

3.2 Algorithm:

Input: G_0 :Seed Profile ,q:Query, δ:Privacy Threshold
Output: Generalized Profile G_* from GreedyIL(H,q, δ)
Assumes:
Q: Generate IL priority Queue
i: iteration index, initialized to 0
t: topics

Steps:

1) Find the Discriminating Power(DP) of Query and Repository=(Profile Granularity +Topic Similarity)/(Expected IC of Topics) and if Discriminating Power(DP) of Query and Repository is less than μ then
2) Find the Profile Granularity of Query and Repository=Summation of (Pr(t|q,G)*IC(t))-H(t|q,G)
3) Find the Topic Similarity of Query and Repository=IC(lca(TG(q)))
4) Find the Information Content IC(t)=log^-1 Pr(t)
5) Find the Pr(t)=Pr(t|root(R))
6) find the Information Loss of each topic
7) find the risk of query and seed profile = Risk(q,G)/Summation of sen(s) ,while Risk(q,G)> δ
8) parent of each topic
9) Process Prune Leaf as $G_i \rightarrow G_{i+1}$ by eliminating –t(topics)
   If t has no sibling then
   Prune leaf only operates on single topic t also insert into Q
   Else If t has sibling then
   Merge t into shadow sibling node
   If No operations on t's sibling in Q then
   Prune leaf only operates on single topic t also insert into Q
   Else Update IL-values for all operations on t's sibling in Q
10) Update i And goto step7
11) Return G_i as G_* goto step 1
12) Return Root(R) as G_*

3.3 Experimental Results:

3.3.1 Accuraccy

According to practical evaluation following graph shows the time required for execution of system as per numbers of Nodes.Figure show that as the numbers of nodes increases the time required for user is also increases.

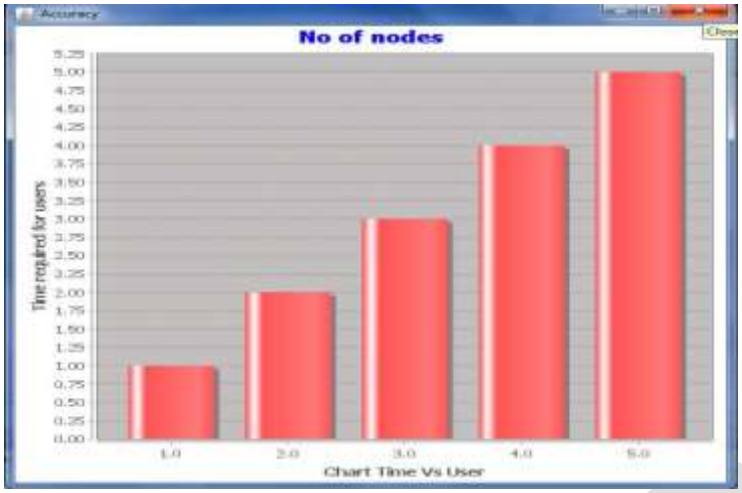

Fig:Time Vs User

5 Key Applications:

.

Personalized web search is considered a promising solution to improve the performance of generic web search. Currently, Google and other web search engines are trying to do personalized search.

6 Future Work and 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. For future work, we will try to resist adversaries with broader background knowledge, such as richer relationships among topics (e.g., exclusiveness, sequentiality, and so on),or capability to capture a series of queries from the Victim. We will also seek more sophisticated methods to build the user profile, and better metrics predict the performance (especially the utility) of the UPS.

7 Acknowledgement:

The authors would like thanks to the publishers, researchers for making their resources available and teachers for their guidance. We also thank the college authority for providing the required infrastructure support. Finally, we would like to extend heartfelt gratitude to friends, family members.

REFERENCES:

[1] G. Chen, H. Bai, L. Shou, K. Chen, and Y. Gao, "*Ups: Efficient Privacy Protection, in Personalized Web Search*," Proc. 34th Int'l ACM SIGIR Conf. Research and Development in Information, pp. 615- 624, 2011.
[2] J. Conrath, "*Semantic Similarity based on Corpus Statistics and Lexical Taxonomy*," Proc. Int'l Conf. Research Computational Linguistics (ROCLING X), 1997.
[3] D. Xing, G.-R. Xue, Q. Yang, and Y. Yu, "*Deep Classifier: Automatically Categorizing Search Results into Large-Scale Hierarchies*," Proc. Int'l Conf. Web Search and Data Mining (WSDM), pp. 139-148, 2008.
[4] Y. Zhu, L. Xiong, and C. Verdery, "*Anonym zing User Profiles for Personalized Web Search*," Proc. 19th Int'l Conf. World Wide Web (WWW), pp. 1225-1226, 2010.
[5] J. Castellı´-Roca, A. Viejo, and J. Herrera-Joancomartı´, "*Preserving User's Privacy in Web Search Engines,*" Computer Comm., vol. 32, no. 13/14, pp. 1541-1551, 2009.
[6] A. Viejo and J. Castell_a-Roca, "*Using Social Networks to Distort Users' Profiles Generated by Web Search Engines*," Computer Networks, vol. 54, no. 9, pp. 1343-1357, 2010.
[7] X. Shen, B. Tan, and C. Zhai, "*Privacy Protection, in Personalized Search*," SIGIR Forum, vol. 41, no. 1, pp. 4-17, 2007