International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169
Volume: 4 Issue: 6 278 - 281

Preserving Privacy for User Profiling in Personalized Web Search

Neha Dewangan
PG Student, Department of Computer Engineering
Alard College of Engineering and Management
Savitribai Phule Pune University
e-mail: dewangan.neha@gmail.com

Prof. Rugraj
Professor, Department of Computer Engineering.
Alard College of Engineering and Management
Savitribai Phule Pune University
e-mail: rugraj@gmail.com

***Abstract*—** As the internet content is growing exponentially, the users of search providers demand their search result to be accurate as per their requirement. In such case Personalized Web Search is one of the options available to the user that present search result as per the users information available in the form of user profile. The major barrier for Personalized Web Search is the unwillingness of user to share their personal information. All the personal information of user is collected during search process and a hierarchical profile based on users preference is created. We propose a client side framework which can be adapted by any PWS that creates users profile on the client side and respect users privacy specified by user during the search process. Also, the generalizing algorithm used during search process for generalizing user profile is discussed in this paper.

Keywords- *Personalized Web Search; user profile; Privacy; risk;*

*****

I. INTRODUCTION

The web Search engine is one of the very important and popular tools used today for seeking information on the internet. Since the content in internet has grown in multiples, at times users experience failure when an irrelevant search result of user query is returned from the search engine. A major problem in existing search engines is they are not adaptive to individual users and same result is displayed to every user with same query. For example: different user can use same query in different search context (eg: java; it may mean Java Island in Indonesia or Java Programming Language). But normal search engine returns same result.

Hence, without knowledge of user information and search context, it is not possible to know Java is used in which sense in a query. Therefore, we must collect user information and personalize user query accordingly for each user. The meaning of Java may be obtained by collecting more information about user and analyzing it as whether a user is a computer science student as against Travel agent or user has bookmarked web pages related to programming language or previous query fired by user is OOPs against cheap flight tickets. Ranking of results followed by above information is very appealing to user as it do not require any extra effort from user. Thus, a general category of search engine called Personalized web search (PWS) is used in order to provide better search result of user query based on individual users need. In personalized web search, user information is collected and analyzed in order to find intention behind issued query fired by user. There are two types of solutions given for Personalized web search, namely click-log based methods and the profilebased methods. The click-log based methods are simple; they simply collect information based on the links users click for viewing the content. It has a strong limitation that it works only on repeated queries. In contrast profile-based method creates users profile based on the users search history. The profile based method has been considered more effective for creating user profile. The profile of user is created based on information gathered implicitly from query history, click through data, users documents, browsing history, bookmarks and so forth. Unfortunately such information collected reveal complete information of users private life. There is no method proposed to provide privacy to the user information. Due to lack of protection of such private information, Personalized web search is not used widely.

A. Paper Organization

The Papers is organized as follows: Section II reviews Literature Survey and Motivation. Section III describes the proposed system architecture. Work done is presented in section IV. And Conclusion is presented in section V.

II. LITERATURE SURVEY

2] States that although personalization has been studied for many years and many techniques have been proposed for it but it is not yet clear whether personalization is effective for different users for different queries. In this paper this problem is studied and some preliminary conclusions are made for it

4] In order to provide personalized search result, user profile or description of users interest is gathered using proxy servers or desktop bots which captures user activities on personal computer

7] Long term search history has rich information regarding users search preferences. This can be used as search criteria to improve retrieval performance.

8] Web search engines are critical for overcoming information overhead. A major problem of such web search engines or information retrieval systems are they dont have methods to create user profile for each individual user and thus results in non optimal retrieval performance.

For PWS there are two classes of privacy protection problem. In one class identification of individual is considered as privacy and the other which considers the sensitivity of data in profile. Class one try to solve problem for pseudoidentity, group identity, no identity, no personal information. Solution

for pseudoidentity proved fragile [12], and no identity and no personal information solutions are expensive and impractical. Thus only solution for no identity is focused. There are many ways for representing a profile. Few among them are bag of words [3], lists/vectors [6]. Recent work uses hierarchical structure. The majority of the hierarchical representations are constructed with existing weighted topic hierarchy/graph, such as ODP [2], [15], [4], [16], Wikipedia [17], [18], and so on. Another work in [11] builds the hierarchical profile automatically via term-frequency analysis on the user data. No focus on implementation of user profiles is considered. It can adopt any hierarchical representation.

To measure performance researchers have proposed metrics for personalized web search which reply on clicking decisions, Rank Scoring [14], Average Precision [20] [11]. In our paper Average Precision has been used to measure performance of personalization. We also use personalization utility and privacy risk as predictive metrics.

A. *Motivation*

Researchers have considered two effects to protect user privacy. One is to improve quality of search and other is to conceal privacy contents of the users profile. Previous studies [11], [13] states people compromise privacy if they obtain better search result for the query. But if only a small part of user profile which is less sensitive is exposed proves to be very useful, as the privacy is maintained. This profile concealing the privacy contents are known as generalized profile. The previous work of privacy is far from Optimal for PWS. The problems observed are stated below:

- No runtime Profiling is supported in existing system. User profile is created once and is used for all queries for the same user. Such strategy has few drawbacks. One of them found in [2] states it doesnt help to improve search quality for ad-hoc queries. Better way is to decide online whether personalization is required and up to what extent profile should be generalized.
- Also, no option available for customization of privacy requirement. That is the privacy level for each individual should differ from one another. Example, in [11], sensitive topics are identified by surprisal metric, which assumes information with less document support are more sensitive. This assumption can be challenged as: suppose user has more documents about financial statement, surprisal of this topic states financial statement is not sensitive and general, which in turn is not true. Such problems are not been addressed yet.
- Till now, continuous user interaction is required for personalizing search results. Thus, we need to minimize this effect.

III. PROPOSED CLIENT-SIDE ARCHITECHTURE FOR PERSONALISED WEB SEARCH

All the mentioned problems have been resolved in framework presented below. The below architecture can be used by any PWS. This basically comprise of client (user) and a Server (search Engine).

As shown in fig1: It consists of a search engine, user and in client side a database is attached which contains complete

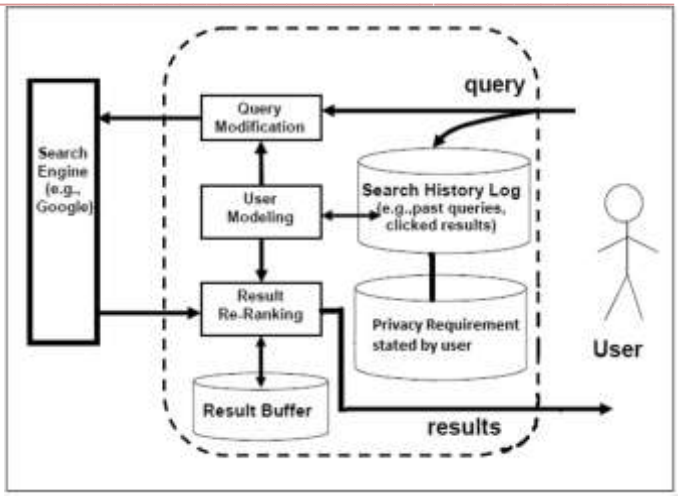

Fig. 1: Client Side Architecture of Personalised Web Search.

information of user profile obtained by search history log i.e. past query information and clicked result. It also contains privacy preference stated by user. It works as follows:

- User fires a query.
- The user query is attached with the user profile maintaining the privacy preference stated by the user. This user profile satisfying privacy requirements stated by user is called Generalized Profile. This generalized profile and the query is then sent to the server
- Search results are personalized with respect to query and generalized profile .
- The personalized results are delivered back to the client.
- The result can be re-ranked based on user profile or presented as such to the user.

The above architecture has three important components. (1) The User modeling creates user profile based on the search history log of the user and also respecting the privacy preference stated by the user. (2) The query modification module formulates the query according to current user module. (3) The result reranking module reranks search results as per users action on clicked web pages. For example: when user clicks on search result, it assumes user is interested in viewing similar topics and hence it reranks the unseen search result. It pulls up information related to view web pages in the top and pushes other related web pages to bottom. Thus, whenever Back or Next link is clicked, the new result would appear which is not same as original search result.

A. *Problem Definition*

First we define how a user profile is created. Then next step is to preserve user privacy requirements on the user profile and to create generalized user profile.

B. *User Profile*

The user profile created will be based on hierarchal structure. The profile will be generated based on public available repository. All personal documents like emails, browsing history etc will be considered as data source for profile. Our assumption is frequently appearing terms are regarded as topic of users interest. In hierarchy terms with higher frequency are kept at top levels while with low frequency are kept at bottom levels. D is regarded as collection

of all user documents whereas D(t) represents the terms in documents. minsup is minimum number of documents in which frequent terms needs to occur. In order to form user profile containing frequent terms, relationship between the frequent terms needs to be considered. Assume two terms ta and tb. There are two types of relations defined for both the terms:

- Similarity: If there is heavy overlap of terms in document set then these terms are considered as similar terms.
- Parent-Child : some specific terms occur together in general terms but reverse is not the case. Eg: badminton occur with sport, but sport can occur with baseball, cricket but not with badminton. Thus tb is child of ta. Thus, using above two rules our algorithm builds hierarchical user profile in top-down manner. Before the algorithm is run on documents all the stop words are removed and stemming is performed. Each document is then treated as list of words. Example: In above Fig 2, 10 documents are available as Data Source based on which user profile is created.

D1: Sports Badminton
D2: ronaldo, sports, socccer
D3: picture, playboy, sex
D4: soccer, sports
D5: AI, algorithm, research
D6:personalized,research,adaptive
D7: channel, sports, sex
D8: search,MSN
D9:AI, research, neuro
D10: google, search, personalized

Fig 2: Data Source Document

C. *Mathematical Model*

User Profile is based on publicly available repository where each topic is associated with repository support supR(t). Therefore, for each topic we have

$$sup_R(t) = \sum sup_R(t') \quad (1)$$

The conditional probability Pr(t|s) where s is ancestor of t is given by repository support.

$$Pr(t \mid s) = sup_R(t) / sup_R(s) \quad (2)$$

Thus, Pr(t) can be further defined to

$$Pr(t) = Pr(t|root(R)) \quad (3)$$

The user is requested to specify a sensitive value for node sen(s)>0 and cost layer is

$$Cost(t) = \sum cost(t') * Pr(t'|t) \quad (4)$$

Algorithm: Pass(n,S(t), minsup, δ)
Input: a node n, term t, supporting documents S(t), thresholds minsup and δ

1. Generate frequent term list {ti} with D(ti) ⩾ minsup sorted by the descending order of frequency.
2. for every term ti:
3. if Sim(ti,tk) > δ , where k < i,
4. set the label as ti/tk, and S(ti/tk) = S(tk) ∪ (ti)
5. else if P(tk—ti) > δ , where k < i,
6. keep the label as tk, and S(tk) = S(tk) ∪ (ti)
7. else
8. generate a new node with label ti, and S(ti) = D(ti)
9. calculate Sup(ti) for every node with label ti, and sort them in a descending order

Algorithm: CreateProf(n, D, minsup,δ)
Input: a node n, supporting document D, thresholds minsup and δ
Output: A user profile U
1. Pass(n, D, minsup, δ)
2. for each child ci labeled ti of node n:
3. CreateProf(ci, S(ti), minsup,δ)

D. *User Defined Privacy Requirement*

Privacy requirements can be specified as sensitive nodes upon whose disclosure, user profile is at privacy risk.

Privacy requirement varies from user to user. One user may consider one topic as private while the other user may not consider the same topic as private in his profile. Therefore, in order to address this difference of privacy, we ask user to specify sensitivity value to each node of created profile. This sensitivity is a positive value that describes how sensitive is that particular topic. We also associate a cost value equivalent to sensitive value which specifies how much loss will be caused by leakage of that topic.

E. *Generalized User Profile*

The term Generalized User Profile is used for profile which exposes only limited profile information maintaining privacy preference specified by user during query search. In general the nodes which are specified as sensitive by the user are forbidden during search process. The process is known as Generalization and the profile formed is known as Generalized profile.

The generalization technique used is mentioned in [1]. Two metrics are used to measure generalized profile. Metric of Utility is used for measuring quality of result of search. Metric of Privacy is used to measure the degree of privacy specified on the user profile.

There are two algorithms mentioned in [1] for Generalizing user profile GreedyDP and GreedyIL algorithm. Greedy algorithm is one which tries to give optimal solution. But the solution obtained is not guaranteed to be always optimal but close to optimal.

GreedyDP algorithm acts in Bottom up manner. It starts with leaf node and for each iteration, chooses leaf topic for pruning. While iterating it maintains best profile-so-far satisfying risk constraint. Iteration is stopped when root topic is reached. Best profile-so-far maintained is final result. It requires recomputation of profiles which increases merory requirement and computational cost.

GreedyIL algorithm improves generalization efficiency. It maintains a priority queue for candidate prune leaf operator in descending order. This lowers the computational cost. GreedyIL terminate the iteration when Risk is satisfied or when there is a single leaf left. Since, there is low computational cost compared to GreedyDP, GreedyIL outperforms GreedyDP.

IV. WORK DONE

In this section practical environments, scenarios, etc used are discussed.

A. *Harware Software Configuration*

Hardware Requirements:
Processor - Pentium IV
Speed - 1.1 Ghz
RAM - 256 MB(min)
Hard Disk - 20 GB
Key Board - Standard Windows Keyboard
Mouse - Two or Three Button Mouse
Monitor - SVGA

Software Requirements:
Operating System - Windows XP
Programming Language - JAVA/J2EE
Java Version - JDK 1.6 & above.
IDE – Eclipse
Database - My SQL

B. *Result*

When the user logs out of the application, the log file created is deleted. Hence there are no histories available of user search which in turn increases the privacy of user search. Only admin have rights to upload all the contents in web. Admin enters the Url Category of content and Location for storing it. Also, the sensitivity nodes are saved, so the privacy is not compromised

V. CONCLUSION

A client side Privacy protection framework is presented in this paper which can be adopted by any Personalized web search to create user profile in hierarchical fashion. Our framework also allows users to impose privacy requirement thorugh which the sensitive information of the user is kept personal without compromising search quality. User profile is created by various levels using Pass and CreateProf algorithm. It also implements GreedyDp and GreedyIL algorithm to create generalized user profile.

ACKNOWLEDGMENT

I would like to thanks for the constant support and guidance I received from my guide Prof. Rugraj. I would also extend my gratefulness to all the faculties who helped me cleared about all the concepts related to my paper.

REFERENCES

[1] Lidan Shou, He Bai, Key Chen and Gang Chen, Supporting Privacy Protection in personalized web search, IEEE Transactions on Knowledge and Data Engineering, 2014.
[2] Z. Dou, R. Song, and J.-R. Wen, A Large-Scale Evaluation and Analysis of Personalized Search Strategies, Proc. Intl Conf. World Wide Web (WWW), pp. 581-590, 2007.
[3] J. Teevan, S.T. Dumais, and E. Horvitz, Personalizing Search via Automated Analysis of Interests and Activities, Proc. 28th Ann. Intl ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR), pp. 449-456, 2005.
[4] M. Spertta and S. Gach, Personalizing Search Based on User Search Histories, Proc. IEEE/WIC/ACM Intl Conf. Web Intelligence (WI), 2005.
[5] B. Tan, X. Shen, and C. Zhai, Mining Long-Term Search History to Improve Search Accuracy, Proc. ACM SIGKDD Intl Conf. Knowledge Discovery and Data Mining (KDD), 2006.
[6] K. Sugiyama, K. Hatano, and M. Yoshikawa, Adaptive Web Search Based on User Profile Constructed without any Effort from Users, Proc. 13th Intl Conf. World Wide Web (WWW), 2004.
[7] X. Shen, B. Tan, and C. Zhai, Implicit User Modeling for Personalized Search, Proc. 14th ACM Intl Conf. Information and Knowledge Management (CIKM), 2005.
[8] X. Shen, B. Tan, and C. Zhai, Context-Sensitive Information Retrieval Using Implicit Feedback, Proc. 28th Ann. Intl ACM SIGIR Conf. Research and Development Information Retrieval (SIGIR), 2005.
[9] F. Qiu and J. Cho, Automatic Identification of User Interest for Personalized Search, Proc. 15th Intl Conf. World Wide Web (WWW), pp. 727-736, 2006.
[10] J. Pitkow, H. Schu tze, T. Cass, R. Cooley, D. Turnbull, A. Edmonds, E. Adar, and T. Breuel, Personalized Search, Comm. ACM, vol. 45, no. 9, pp. 50-55, 2002.
[11] Y. Xu, K. Wang, B. Zhang, and Z. Chen, Privacy-Enhancing Personalized Web Search, Proc. 16th Intl Conf. World Wide Web (WWW), pp. 591-600, 2007. [12] K. Hafner, Researchers Yearn to Use AOL Logs, but The
[12] Y Hesitate, New York Times, Aug. 2006.
[13] A. Krause and E. Horvitz, A Utility-Theoretic Approach to Privacy in Online Services, J. Artificial Intelligence Research, vol. 39, pp. 633-662, 2010.
[14] J.S. Breese, D. Heckerman, and C.M. Kadie, Empirical Analysis of Predictive Algorithms for Collaborative Filtering, Proc. 14th Conf. Uncertainty in Artificial Intelligence (UAI), pp. 43-52, 1998.
[15] P.A. Chirita, W. Nejdl, R. Paiu, and C. Kohlschu tter, Using ODP Metadata to Personalize Search, Proc. 28th Ann. Intl ACM SIGIR Conf. Research and Development Information Retrieval (SIGIR), 2005.
[16] A. Pretschner and S. Gauch, Ontology-Based Personalized Search and Browsing, Proc. IEEE 11th Intl Conf. Tools with Artificial Intelligence (ICTAI 99), 1999.
[17] E.Gabrilovich and S. Markovich, Overcoming the Brittleness Bottleneck Using Wikipedia: Enhancing Text Categorization with Encyclopedic Knowledge, Proc. 21st Natl Conf. Artificial Intelligence (AAAI), 2006.
[18] K. Ramanathan, J. Giraudi, and A. Gupta, Creating Hierarchical User Profiles Using Wikipedia, HP Labs, 2008.
[19] K. Jarvelin and J. Kekalainen, IR Evaluation Methods for Retrieving Highly Relevant Documents, Proc. 23rd Ann. Intl ACM SIGIR Conf. Research and Development Information Retrieval (SIGIR), pp. 41-48, 2000.
[20] R. Baeza-Yates and B. Ribeiro-Neto, Modern Information Retrieval. Addison Wesley Longman, 1999.
[21] X. Shen, B. Tan, and C. Zhai, Privacy Protection in Personalized Search, SIGIR Forum, vol. 41, no. 1, pp. 4-17, 2007.
[22] Y. Zhu, L. Xiong, and C. Verdery, Anonymizing User Profiles for Personalized Web Search, Proc. 19th Intl Conf. World Wide Web (WWW), pp. 1225-1226, 2010.
[23] Y. Xu, K. Wang, G. Yang, and A.W.-C. Fu, Online Anonymity for Personalized Web Services, Proc. 18th ACM Conf. Information
[24] J. Castell-Roca, A. Viejo, and J. Herrera-Joancomart, Preserving Users Privacy in Web Search Engines, Computer Comm., vol. 32, no. 13/14, pp. 1541-1551, 2009.