International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 3, Issue 01, January 2016
Available at http://internationaljournalofresearch.org

User Customized Privacy Protection in Personalized Web Search

Author1:
Kenche Vamshi Krishna
M.Tech,Dept of CSE Aurora's Scientific,Technological and Research Academy Hyderabad-500081
vamshi550@gmail.com
Contact: 8008459759.
Author 2:
T. Malathi
Senior Assistance Professor,Dept of CSE Aurora's Scientific,Technological and Research Academy, Hyderabad-500081.
malathibhuvan@gmail.com

ABSTRACT:

Personalized web search has been introduced to enhance the user experience in faster decision making by neglecting the least relevant web search results for user. At the same time users do not want their personal information to be revealed to the outside world. User's disinclination to tell their personal information during search has becomes a major barricade for the wide build-up of personalized web search. Achieving the greater privacy along with the personalization is big challenge where previous researches could not able to achieve to the complete extent. This paper discusses privacy protection in personalized web search applications that represents userdesire as taxonomy user profiles. Generalize profile by queries while reference user specified a private requirementusing a personalized web search framework called User Customizable Privacy Preserving Search (ups). The UPS framework is a for step process. They are generating the user profile, privacy requirement customization, mapping the query topic with the corresponding domain and runtime profile generalization. And also in this paper we study how the two predictive metrics personalized and privacy protection is achieved with the help of two algorithms namely Greedy Discriminating Power and Greedy Information Loss algorithms respectively.

Key Words: Privacy protection; Profile generalization; privacy requirement customization; personalized search; privacy risk; Search engines.

1. INTRODUCTION:

The importance of accessing the best possible personalized results from the web is rapidly growing among the users. The users of large scale organisations viz., e-commerce that maintains huge repository or their personal data and the users of internet generally have a lot of interest towards getting the more personalized results and to avoid to the best possible extent those results which are no more relevant to their search criteria. Such irrelevance is due to the enormous variety of users search criteria. The process of providing the better search results which are tailored for individual user needs is called as personalized web search results.

But, user has to invest his or her personal information like queries or0their topics of interest in order to get personalized results. There are two important aspects here. One, the user should get more personalized results. second, user's personal information should be preserved without revealing to the outside world.

1.1. Click-log method:

There were many research have been carried out to achieve the above two aspects. But the results are very far from the optimal. One solution using user's browsing history. This mechanism is entitled as personalized web search results using click-log. Still there are so many disadvantages associated with click log based method. though the click log based method has achieved best results to some extent in personalized web search but is a complete fail in privacy protection of the user.

The main drawbacks of the click log based method are

a) The privacy of the user is of no consideration.

b) User's browser history is not a reference for complete user profile of interests.

As a result, the above method could not able to solve the two problems discussed earlier.

1.2 Bookmarks:

The personalized web search is also achieved with the help of user bookmarks. Though it cannot solve the complete purpose, but to some extent it can personalize the search results. If the user has not yet bookmarked any topic yet, the question of personalized web search will be under threat. And also, such implicitly collected personal data can easily reveal the user's private life. As a result privacy cannot be maintained. This is the major drawback of this procedure.

1.3. Motivations: The users of internet have grown enormously in the recent days. Generally users have a tendency towards accessing the more personalized search results. Sometimes they even compromise their personal user profile said that if the results are more personalized. Here researchers have to consider two contradictory aspects. One, to improve the search quality with the personalization of the search results and second, was hiding the user private information from all privacy risks like eaves dropping etc. As these two are contradictory to each other, to achieve one aspect, the second has to be compromised and vice versa. But in general there is a trade off between the search quality and the level of privacy protection. Unfortunately, the previous works of privacy preservingPWS are far from optimal. The problems with the existingmethods are explained in the following observations:

1. The existing profile-based PWS do not support runtime profiling.

2. The existing methods do not take into account thecustomization of privacy requirements. This probablymakes some user privacy to be overprotected while others insufficiently protected.

3. Many personalization techniques require iterative userinteractions when creatingpersonalized search results.

2. RELATED WORKS:

Here we focus on the related works of profile-basedpersonalization and privacy protection in PersonalizedWeb Search system.

2.1 Profiles-Based Personalization:

The previous works on Profile-based Personalized WebSearch mainly focuses on improving the search utility.Generally the Profile-based Personalized Web Searchprovides the search results by referring to the userprofile that reveals an individual information need. Herewe review the previous solutions to PWS on two aspects,namely the representation of profiles, and the measure ofthe effectiveness of personalization.To facilitate different personalization strategiesmany profile representations are available in theliterature. However in most recent the user profiles arebuilt in hierarchical structures due to their strongerdescriptive ability, better scalability, and higher accessefficiency. Mostly the hierarchical representations areconstructed with existing weighted topichierarchy/graph, such as ODP, Wikipedia and so on.

Another technique is to build the hierarchical profileautomatically via term-frequency analysis on the userdata. In our proposed UPS framework, we do not focuson the implementation of the user profiles. Actually, ourframework can potentially adopt any hierarchicalrepresentation based on taxonomy of knowledge.

For the performance measures of PWS in theliterature, Normalized Discounted Cumulative Gainis a common measure of the effectiveness of aninformation retrieval system. But there is a lot of humaninvolvement in performance measuring and to reducethis researchers also propose other metrics ofpersonalized web search that rely on clicking decisions,including Average Precision (AP), Rank Scoring, andAverage Rank. In our framework we use the AveragePrecision metric, proposed by Dou et al., to measure theeffectiveness of the personalization in UPS.Our work also proposes two predictive metrics,namely personalization utility and privacy risk, on aprofile instance without requesting for user feedback.

2.2 Privacy Protection in PWS System

There are two classes of privacy protection problems forPWS. One class includes which treat privacy as theidentification of an individual. The other includes whichconsider thesensitivity of the data, particularly the userprofiles, exposed to the PWS server. In the literature ofprotecting user identifications (class one) we try to solvethe privacy problem on different levels, including thepseudo identity, the group identity, no identity, and nopersonal information. The Solution for the first level isproved too fragile. The third and fourth levels areimpractical due to high cost in communication andcryptography. Therefore, the existing efforts focus on thesecond level. The solutions in class two do not requirethird-party assistance or collaborations between socialnetwork entries. In these solutions, users only trustthemselves and do not allow the exposure of theircomplete profiles to an anonymity server.Krause and Horvitz and Xu et al. proposed a privacyprotection solution for PWS but unfortunately, this workdoes not address the query utility, which is crucial for the service quality of PWS. But our approach takes boththe privacy requirement and the query utility intoaccount. We also provide personalized privacy protection in PWS. In this approach we allow users tocustomize privacy needs in their hierarchical userprofiles. Another problem that concerns the privacyprotection in PWS is that personalization may havedifferent effects on different queries. Queries withsmaller click-entropies, namely distinct queries, are expected to benefit more from personalization, whilethose with larger values (ambiguous ones) are not andthis may even cause privacy disclosure. In our UPSframework, we differentiate distinct queries fromambiguous ones based on a client-side solution using thepredictive query utility metric. In this paper, we extendand detail the implementation of UPS and also the metricof personalization utility to capture our three newobservations and they are:

1. The existing profile-basedPWS do not support runtime profiling.
2. The existingmethods do not take into account the customization of privacy requirements.
3. Many personalizationtechniques require iterative user interactions whencreating personalized search results. We also propose anew profile generalization algorithm called GreedyIL.Based on three observations newly added in theextensions, the efficiency and

International Journal of Research
International Journal of Research (IJR)
e-ISSN: 2348-6848, p- ISSN: 2348-795X Volume 3, Issue 01, January 2016
Available at http://internationaljournalofresearch.org

stability of the newalgorithm outperforms the old one significantly.

3. PROPOSED SYSTEM:

The above problems are addressed in our UPS(literally for User customizable Privacy-preservingSearch) framework. UPS is distinguished fromconventional PWS in that it

1) Provides runtimeprofiling, which in effect optimizes the personalizationutility while respecting user's privacy requirements;
2) Allows for customization of privacy needs; and
3)Does not require iterative user interaction.

SYSTEM ARCHITECTURE:

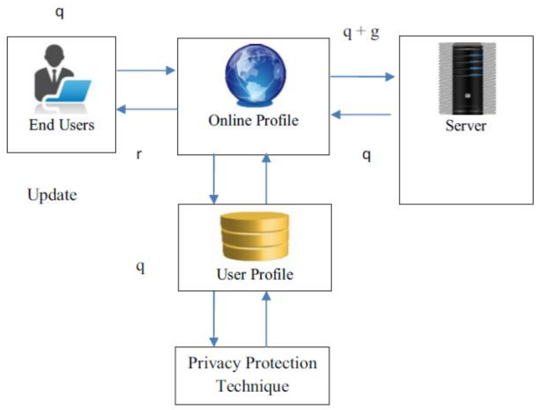
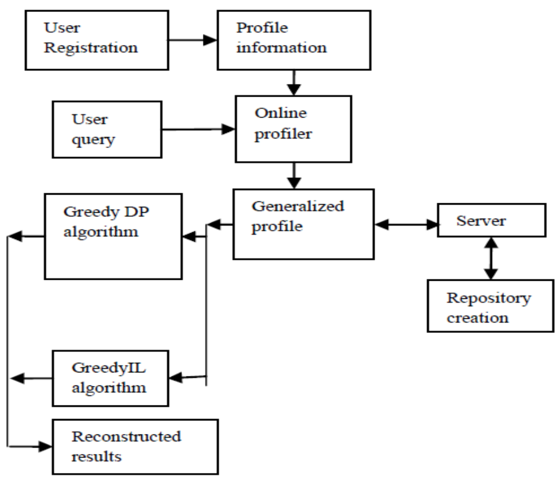


Figure: System Architecture of UPS.

Step by step detailed procedure in ups frame work:



As illustrated in Fig., UPS consists of anontrusty search engine server and a number of clients.

Each client (user) accessing the search service trusts noone but himself/herself.The framework works in two phases, namely theoffline and online phase, for each user. During the offline phase, a hierarchical user profile is constructedand customized with the user-specified privacyrequirements. The online phase handles queries asfollows:

1. When a user issues a query *qi* on the client, theproxy generates a user profile in runtime in thelight of query terms. The output of this step is ageneralized user profile *Gi*satisfying the privacy requirements. The generalization process is guidedby considering two conflicting metrics, namely thepersonalization utility and the privacy risk, bothdefined for user profiles.
2. Subsequently, the query and the generalized userprofile are sent together to the PWS server forpersonalized search.
3. The search results are personalized with theprofile and delivered back to the query proxy.
4. Finally, the proxy either presents the raw results tothe user, or reranks them with the complete userprofile.

Specifically, each user has to undertake thefollowing procedures in our solution:

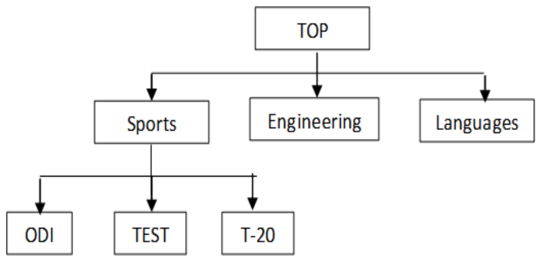
1. Constructing the user profile
2. Customization of user's privacy requirements
3. Mapping the query topic to its is corresponding domain
4. Profile Generalization

Phase-1:Constructing the User Profile:

Step-1: The user profile is a repository of 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 tree *subtr(t, R)* as the taxonomy accompanying t.

Step-2: In addition, each topic *t*∈*R* is associated with arepository support, denoted by *supR(t)*, which quantify show often the respective topic is touched in humanknowledge.

Step-3: If the support values are notavailable. Then *supR(t)* can be calculated as the count of leaves in *subtr(t, R)*.



Phase-2: Customization of User's Privacy Requirements:

In this phase, the privacy requirements of the user, such as which details they would like to reveal and which details they would not like to reveal and how it can be accomplished is discussed. Customization of user's privacy requirements depends upon the sensitive values of the topics. From users perspective, the sensitivity of the topics differs from one topic to another. So, to address the difference in privacy concerns, we allow the user to specify a sensitivity value for every node. This is denoted as sen(S).

Phase-3: User Profile Generalization:

Method of 'Forbidding':

Since the sensitivity values explicitly denotes the users privacy concerns, so the most straight forward way to preserve the user's privacy is to remove the sub trees nodes at all sensitive nodes.

Problem with 'Forbidding':

The method of forbidding has certain disadvantages. The problem with forbidding is though the nodes with sensitive values are forbidden, it cannot guarantee the privacy. Because, third party attacker or eaves dropping attack who could be able to access the user profile, can predict the forbidden nodes depending on their siblings.

In the above taxonomy of topics, the node with 'Test Cricket' is a sensitive node from the users' perspective. It means he would not like to reveal this key word to the outside world. So, according to the method of forbidding, this node (Test Cricket) should be forbidden. But simply forbidding this node will not solve the problem. Because, this node van easily be predicted with the help of its siblings.

Solution of the forbidding problem:

The problem of forbidding is resolved with the help of Greedy Information Loss algorithm.

Greedy information loss algorithm:

To avoid the risk of forbidding, the method to be undertaken is to detect and remove a set of nodes such that privacy risk introduced by exposing the sub tree is always under control.

If the sensitivity of nodes is less i.e. nodes with low sensitivity, it is unnecessary to remove them. Since, the problem with those nodes is almost negligible.

Coming to the nodes with high sensitive values, instead of removing only the sensitive node, the complete sub tree has to be removed and is moved to a separate shadow data structure.

Step by Step Procedure:

Step-1: Identify the nodes with high sensitivity and low sensitivity values using a threshold limit value.

Step-2: Detect and remove a set of nodes such that risk introduced by exposing the sub tree is always under control.

Step-3: Low sensitivity nodes are unnecessary to remove since the privacy risk introduced by exposing those nodes is always under control.

Step-4: The procedure of low information loss for the nodes with high sensitivity nodes is as follows.

If the sensitive value is greater than the threshold value i.e risk (q, Gi) > T, prune the leaf from the sub tree and move the node to a set S.

Step-5: If G' is a profile obtained by applying a prune leaf operation on G, then DP(q; G) ≥ DP(q, G').

Step-6: Specifically, each candidate operator in the queue is a tuple like op = (t, IL (t, Gi)), where t is the leaf to be pruned by op and IL (t, Gi), indicates the IL incurred by pruning t from Gi.

Step-7: The iterative process can terminate whenever ϑ- risk is satisfied.

Step-8: The second term (TS(q, G) remains unchanged for any pruning operations until a single leaf is left (in such case the only choice for pruning is the single leaf itself).

Step-9: In C1, t is a node with no siblings, and In C2, t is a node with siblings. The case C1 is easy to handle. However, the evaluation of IL in case C2 requires introducing a shadow sibling of t.

Step-10: Each time if we attempt to prune t, we actually merge t into shadow to obtain a new shadow leaf shadow0, together with the preference of t,

Step-11: Prune-leaf only operates on a single topic t. Thus, it does not impact the IL of other candidate operators in Q. While in case C2, pruning t incurs recomputation of the preference values of its sibling nodes.

Step-12: Once a leaf topic t is pruned, only the candidate operators pruning t's sibling topics need to be updated in Q. In general, Greedy IL traces the information loss insteadof the discriminating power. This saves a lot of computational cost.

Phase-4: Mapping The Query Topic With The Corresponding Domain:

- Given a query q, the purposes of query-topic mapping are
  1) to compute a rooted sub tree of H, which is called a seed profile, so that all topics relevant to q are contained in it; and
  2) to obtain the preference values between q and all topics in H.
- This procedure is performed in the following steps: 1. Find the topics in R that are relevant to q. We develop an efficient method to compute the relevance's of all topics in R with q.
- These values can be used to obtain a set of non-overlapping relevant topics denoted by T(q), namely the relevant set.
- We require these topics to be non-overlapping so that T(q), together with all their ancestor nodes in R, comprise a query-relevant tree denoted as R(q).
- Apparently, T(q) are the leaf nodes of R(q). Note that R(q) is usually a small fraction of R.
- 2. Overlap R(q) with H to obtain the seed profile G0, which is also a rooted sub tree of H. For example, by applying the mapping procedure on query "Eagles," it obtain a relevant set T(Eagles).
- The sample profile with its query-relevant tree R(Eagles) gives the seed profile Gb, whose size is significantly reduced compared to the original profile. The leaves of the seed profile G0 (generated from the second step) form a particularly interesting node set the overlap between set T(q) and H.
- We denote it by TH(q), and obviously we have TH(q) is a subset of T(q).

- Then, the preference value of a topic t is element of H is computed.
- Though this probability is not used in this procedure, it is needed to evaluate the discriminating power of q, and to decide whether to personalize a query or not.

Greedy Discriminating Power:

Greedy Discriminating Power algorithm is used in personalization of web search results. This algorithm gives optimal solution hence called a Near Optimal Greedy Algorithm. The purpose of greedy discriminating power algorithm is effectively decided which information to be displayed to the user first and which are to be omitted.

Actually, there are two basic ingredients every greedy algorithm has in common:

- **Greedy Choice Property**: from a local optimum we can reach a global optimum, without having to reconsider the decisions already taken.
- **Optimal Substructure Property**: the optimal solution to a problem can be determined from the optimal solutions to its subproblems.

When to use Greedy Discriminating Power Algorithm:

- A problem that seems extremely complicated on the surface signal a greedy approach.
- Problems with a very large input size (such that a n^2 algorithm is not fast enough) are also more likely to be solved by greedy than by backtracking or dynamic programming.
- Despite the rigor behind them, you should look to the greedy approaches through the eyes of a detective, not with the glasses of a mathematician.

NP-hardness (*n*on-deterministic *p*olynomial-time hard), in computational complexity theory, is a class of problems that are, informally, "at least as hard as the hardest problems in NP". More precisely, a problem *H* is NP-hard when every problem *L* in NP can be reduced in polynomial time to *H*. As a consequence, finding a polynomial algorithm to solve any NP-hard problem would give polynomial algorithms for all the problems in NP, which is unlikely as many of them are considered hard.

Approach: Making the locally optimal choice at each stage with the hope of finding a global optimum.

Advantage of Greedy Discriminating Power algorithm is that the solutions to the smaller instances of the problem can be straight forward and easy to understand.

Disadvantages of Greedy Discriminating Power algorithm is, if the results accurate, then performance may be poor. If the performance is poor, the results may not be accurate.

- Accurate Results --- Not fast enough
- Fast enough --- Not accurate results.

Procedure:

Step-1: Map the given query topic with the corresponding domain.

Step-2: Let 'N' be the number of nodes in a domain and {I} is the node from 1 to n.

Step-3: Chose the nodes with the high preference value.

Step-4: Iterate to their sibling nodes with the ascending order of their discriminating power values and also privacy risk values.

Step-5: Now perform the same iteration in every sibling nodes of the current parent node.

Step-6: Obtain the personalized search results.

The advantages Enhanced Privacy Protection Framework is as follows:

- It enhances the stability of the search quality

- Improves the privacy protection against different type of attacks
- It avoids the unnecessary exposure of the user profile
- It provides runtime profiling

CONCLUSION

In this paper we presented a client-side privacy protection framework called UPS (User CustomisablePrivacy Preserving Search) for personalized web search. UPS could likely be adopted by any PWS that captures user profiles in a hierarchical taxonomy. Our proposed framework provided customized privacy requirements via the hierarchical profiles to the users. Through this profile, users' control what portion of their private information is exposed to the server and the users can specify to which degree the content should be protected. In addition, UPS also performed online generalization on user profiles to protect the personal privacy without compromising the search quality. 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 Generalization, with its NP-hardness proved. We proposed 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 proposed 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. The experimental results revealed that while preserving user's customized privacy requirements our proposed UPS framework could achieve quality search results. The results also confirmed the effectiveness and efficiency of our solution. There is a scope in future that we could try to resist adversaries with broader background knowledge, such as richer relationship among topics (e.g., exclusiveness, sequentially, and so on), or capability to capture a series of queries from the victim and would work in future. We will also find more advanced method to build the user profile, and better metrics to predict the performance, especially the utility of UPS.

Future Enhancement

For future work, we will also seek more sophisticated method tobuild the user profile, and better metrics to predict theperformance (especially the utility) of UPS. we can also implement the hierarchical divisive approach for retrieving the search results. It will gives better performance when compared with our proposed System.

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.

[5] K. Sugiyama, K. Hatano, and M. Yoshikawa, “Adaptive Web Search Based on User Profile Constructed without any Effort from Users,” Proc. 13th Int’l Conf. World Wide Web (WWW), 2004.

[6] X. Shen, B. Tan, and C. Zhai, “Implicit User Modeling for Personalized Search,” Proc. 14th ACM Int’l Conf. Information and Knowledge Management (CIKM), 2005.

[7] X. Shen, B. Tan, and C. Zhai, “Context-Sensitive Information Retrieval Using Implicit Feedback,” Proc. 28th Ann. Int’l ACM SIGIR Conf. Research and Development Information Retrieval (SIGIR), 2005.

[8] F. Qiu and J. Cho, “Automatic Identification of User Interest for Personalized Search,” Proc. 15th Int’l Conf. World Wide Web (WWW), pp. 727-736, 2006.

[9] 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.

[10] Y. Xu, K. Wang, B. Zhang, and Z. Chen, “Privacy-Enhancing Personalized Web Search,” Proc. 16th Int’l Conf. World Wide Web (WWW), pp. 591-600, 20070.