INTERNATIONAL JOURNAL FOR ADVANCE RESEARCH IN ENGINEERING AND TECHNOLOGY

WINGS TO YOUR THOUGHTS…..

PRIVACY PROTECTION FOR USER GROUPS IN PERSONALIZED WEB SEARCH WITH SEARCH EFFICIENCY

Greeshma A S[1], Lekshmy P L[2]

[1]Mtech Scholar, Dept. of CSE, LBS Institute of Technology for Women,
Trivandrum, Kerala -695012
greeshmaas2012@gmail.com
[2]Assistant Professor, Dept. of CSE, LBS Institute of Technology for Women,
Trivandrum, Kerala -695012
lekshmyvinod@gmail.com

Abstract: *Personalized web search (PWS) refers to search experiences that are adopted specifically for individual's interests by incorporating information about an individual beyond specific query provided, but at the same time this will raises new privacy challenges. User's hesitation to disclose their private information during search has become major issue on personalization technologies. To overcome this problem privacy protection is required. Personal user information is used to create user profiles, which is a critical data. In this system user profile is modeled as a hierarchical structure, which is created from a taxonomy repository. For privacy protection, Privacy protected PWS framework is proposed. An online profiler is designed in this system, which can adaptively generalize profiles by queries while respecting user specified privacy requirements. Runtime generalization aims at providing search efficiency along with privacy protection of user profiles. To prevent the information loss while performing runtime generalization, MinimizeIL algorithm is proposed. Location based search results can be integrated along with the profile based personalization to retrieve more relevant results, based on a location without specifying the same in the query. A group level privacy can be achieved so that user profiles in a group are secured, that are stored at the client side. With this system, even at offline phase also users can get the viewed results of their friends. Thus search efficiency of the system can also increase by reducing the execution time significantly.*

Keywords: *Generalization, Personalization, User profile, MinimizeIL.*

1. INTRODUCTION

Web is made up of 60 trillion individual pages and it is constantly growing, to find document of our need we follow link from page to page. To deliver best results programs and formulas are written, algorithms look to understand what do you mean, by checking spelling, search methods, synonyms after checking all possible clues most relevant document from index is delivered to the user. People are getting more dependent on WSE's for information needs. People use web search for many reasons like for finding queries of daily need or business issues or for getting information about entities, web search engine sorts information out of millions of pages and send results to the users [1]. Because of large size of web or amount of information continuously increasing user may get thousands of results which may be related or not related i.e queries submitted by different type of user with different need may get same results. The features of the query submitted by the user are; like in complete, short and ambiguous. For example for the query "bat" some users like sports men, cricket lover may be interested in documents. User's like scientist or biology professor may want documents related to "bat bird". If same results are delivered to both the users it will create problems to find the actual content which user wants.

The user clicks one or more documents that look relevant and skips those documents that the user is not interested in. 68% of the users click a search result within the first page of results and 92% click a result within the first three pages. Therefore, WSEs must put the links that are more interesting for the users in the first result page. It is the need to deliver related contents to user based on user profile its challenge when different user search for similar query in different context, in this era of technology users expect WSE to be intelligent and serve results according to their needs where our general search engine failed. The solution is personalized web search (PWS). Personalizing web search (PWS) is a technique which provides better search results according to individual's need. PWS motivates to concentrate more on creating interactive content, high quality content but raises reasonable concerns about privacy. User's hesitation to disclose their private information during search has become major issue on personalization technologies. For example system that are personalize some advertisements according to physical location of user or their search history [13], introduces new privacy challenges that may discourage the wide adoption of personalization technologies [1].

User may be uncomfortable to expose personal information, which lead to being increasingly identifiable and can release personal information of user. Most efforts have ignored privacy to enhance utility, as these are two contradicting effects, to improve search quality user should compromise on search utility or vice versa. For personalizing online services implicit and explicit methods can be used. In explicit personalizing methods users specify there topic of interest that reside on server or client on the other hand implicit personalizing methods user is not aware about his/her information collection which includes user location, clicks and search activities. To capture user's interest for personalization two methods are used namely click based and profile based [2]. Click through is simple, gathers data generated by user click i.e. repeated queries from same user. Profile based method maintains complete user profile to form user interest models; these are effective for all queries by same user [3]. Even though these user profile can't identify users directly but they can achieve identification by recovering IP address linked to bunch of queries or with his/her name, national Id etc. single query might not reveal identity of a user, bunch of queries might cause this situation. An example of this situation is the case of Thelma Arnold, user of the AOL's WSE [4], which was identified by her searches, submitted over a three-month period. All these queries were hidden behind a pseudonym to protect the real identity of the user. However, the aggregation of hundreds of queries was enough to identify and profile her. User profile contains sensitive and personal information which pose serious privacy threat to user. WSE's are not proper to use for privacy instead user should use privacy preserving mechanism to prevent exposing information. Profile-based methods improve the search experience with complicated user-interest models generated from user profiling techniques. Profile-based methods [2] can be potentially effective for almost all sorts of queries. The two contradicting effects during the search process to be considered. Improve the search quality with the personalization utility of the user profile [5] and the need to hide the privacy contents existing in the user profile to place the privacy risk under control. To improve the search efficiency, location of the current user is traced with the use of network service provider and is sent along with query and generalized profile to the server. Search results based on user previous history also can be obtained with this system.

A community of users called friends is created and the friends can share their less sensitive topics. The real intention behind the community is to get search results at the offline phase also. Each user in the community has their privacy and is implemented using generalization method. They can share what they want to share and if they search for query which is previously searched by any other friend in their community, then the user can get search results at offline stage. Without connecting to the network the results viewed by the friends can be available for repeated queries in the community. This can increase the search efficiency of the PWS search engine.

2. LITERATURE REVIEW

2.1 Profile Based Personalization

The basic idea of the existing works is to tailor the search results by referring to, often implicitly, a user profile that reveals an individual information goal. Many profile representations are available in the literature to facilitate different personalization strategies [6]. Important among them are:

- Lists / vectors or bag of words: Earlier techniques utilize term lists/vectors or bag of words to represent their profile. It is the simple representation in information retrieval system. Here a text is represented as the bag of its words, disregarding grammar and even word order but keeping multiplicity. In each vector the other entry will be the count of that word.
- Hierarchical representation: Most recent works build profiles in hierarchical structures due to their stronger descriptive ability, better scalability, and higher access efficiency. The majority of the hierarchical representations are constructed with existing weighted topic hierarchy/graph, such as ODP, Wikipedia, and so on. Hierarchical profile can also build automatically via term-frequency analysis on the user data.

2.2 Privacy Protection in PWS

Generally there are two classes of privacy protection problems for PWS. One class includes those treat privacy as the identification of an individual. The other includes those consider the sensitivity of the data, particularly the user profiles, exposed to the PWS server. The two classes are described below.

2.2.1 Identification of Individual

Typical works in the literature of protecting user identifications (class one) try to solve the privacy problem on different levels [11], including the pseudo-identity, the group identity, no identity, and no personal information. Solution to the first level is proved too fragile. The third and fourth levels are impractical due to high cost in communication and cryptography. Therefore, the existing efforts focus on the second level. The main techniques come under the first class are:

- Online Anonymity: It works based on user profiles by generating a group profile of k users. Using this approach, the linkage between the query and a single user is broken [7].
- Useless User Profile (UUP): This protocol is proposed [8] to shuffle queries among a group of users who issue them. As a result any entity cannot profile a certain individual. These works assume the existence of a trustworthy third-party anonymizer, which is not readily available over the internet at large.
- Legacy Social Networks: Instead of the third party to provide a distorted user profile to the web search engine. In the scheme [9], every user acts as a search agency of his or her neighbors. They can decide to submit the query on behalf of who issued it, or forward it to other neighbors.

2.2.2 Sensitivity of Data

The solutions in class two do not require third-party assistance or collaborations between social network entries. In these solutions, users only trust themselves and cannot tolerate the exposure of their complete profiles an anonymity server. The main techniques come under the first class are:

- Statistical Techniques: To learn a probabilistic model [10], and then use this model to generate the near-optimal partial profile. One main limitation in this work is that it builds the user profile as a finite set of attributes, and the probabilistic model is trained through predefined frequent queries. These assumptions are impractical in the context of PWS.
- Generalized Profiles: Proposed a privacy protection solution for PWS based on hierarchical profiles. Using a user-specified threshold [12], a generalized profile is obtained in effect as a rooted sub tree of the complete profile.

Y. Xu, K. Wang, G. Yang proposed the notion of online anonymity [7] to enable users to issue personalized queries to an un-trusted web service while with their anonymity preserved. Ensures that each query entry in the query log cannot be linked to its sender and an algorithm that achieves online anonymity through the user pool is introduced.

In [8] J. Castelli-Roca, A. Viejo and J. Herrera presents a novel protocol Useless User Profile (UUP) protocol, specially designed to protect the users' privacy in front of web search profiling. System provides a distorted user profile to the web search engine. This scheme also uses cryptographic building blocks such as Elgamal encryption, key generation, message encryption and decryption etc. for effective communication. The main idea of this scheme is that each user who wants to submit a query will not send her query but a query of another user instead. At the same time, her query is submitted by another user. The protocol assumes that, users follow the protocol correctly and no collision happens between entities, but in real it may be not the case.

Y. Zhu, L. Xiong, and C. Verdery et al [16] an optimal privacy notion to bound the prior and posterior probability of associating a user with an individual term in the anonymized user profile set is proposed. The authors proposes a novel bundling technique that clusters user profiles into groups by taking into account the semantic relationships between the terms while satisfying the privacy constraint.

A. Viejo and J. Castellia-Roca, propose a new scheme [9] designed to protect the privacy of the users from a web search engine that tries to profile them. The system uses social networks to provide a distorted user profile to the web search engine. It exploits the existence of neighborhoods of on-line users (social networks). In this way, a user generates queries and she can submit them directly to the WSE or she can forward them to her neighbors in the social network. The proposed system does not create groups for submitting queries.

In [12] X. Xiao and Y. Tao**,** presented a new generalization framework based on the concept of personalized anonymity. This technique performs the minimum generalization for satisfying everybody's requirements, and thus, retains the largest amount of information from the micro-data.

In [17] Y. Xu, K. Wang, B. Zhang et al**,** presents a scalable way for users to automatically build rich user profiles. A significant improvement on search quality can be achieved by only sharing some higher-level user profile information, which is potentially less sensitive than detailed personal information. A search engine wrapper is developed on the server side to incorporate a partial user profile with the results returned from a search engine. Rankings from both partial user profiles and search engine results are combined. The customized results are delivered to the user by the wrapper.

In [14] Lidan Shou and Gang Chen et al, uses hierarchical user structure for modeling user interests. The system provides generalization of user profile with use of an online profiler at the client side. The system is expected to enhance the search efficiency with the personalization utility, along with the privacy protection of user profile contents. PWS framework called UPS (User Privacy Preserving Search) is introduced, which can adaptively generalize profiles by queries while respecting user specified privacy requirements is proposed. 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. For generalization two greedy algorithms, namely GreedyDP and GreedyIL are used. An online prediction mechanism for deciding whether personalizing a query is beneficial or not, is also proposed in this work.

3. PRIVACY PROTECTED PWS FRAMEWORK

The framework proposed 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 Fig. 1, framework consists of a non-trusty 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.

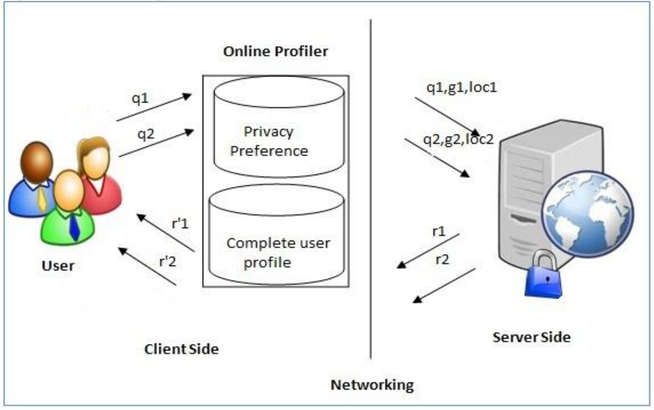

Figure 1: Privacy protected PWS architecture

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 user can create a community or have friends and can share search history at offline phase. The online phase handles queries as follows.

- 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 MinimizeIL algorithm.
- Subsequently, the query and the generalized user profile are sent together to the PWS server for personalized search.
- The search results are personalized with the profile and delivered back to the query proxy.
- Finally, the proxy either presents the raw results to the user, or re-ranks them with the complete user profile. The search results are also based on user's previous history and it also contains the viewed results of user friends if the keyword matches.

4. GENERALIZATION

As illustrated in Figure.1, system architecture consists of a non-trusty search engine server and a number of clients. The main components of the system are User interface, Online Profiler and Search engine server. With the user interface the user can made their preferences (interests) with the available taxonomy repository R, which covers almost the entire human knowledge.ie: the user profile H. The sensitive nodes can be specified in the user profile. Customized privacy requirements [15] can be specified with a number of sensitive-nodes (topics) in the user profile, whose disclosure (to the server) introduces privacy risk to the user.

Each client accessing the search service trusts no one but himself/herself. The key component of the architecture 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 online profiler creates generalized profile on runtime with the given query by the user and is send to the server along with the query. The server provides results back to the online profiler and is given to the client.

The query along with the generalized profile is given to the server. The server returns back the most relevant documents with the given profile. After connected to the network only, the search results can be get from the server. Any server can be used for getting such search results like Bing, Google, Yahoo etc. Free Search services can be getting from Google ajax services. User can create friends ie: a community. Each user in the community has their privacy and is implemented using generalization method. They can share what they want to share and if they search for query which is previously searched by any other friend in their community, then they can also search online. Without connecting to the network the results viewed by the friends can be available for repeated queries in the community. This can increase the search efficiency of the PWS search engine.

User profile H is created from the taxonomy Repository R, which is the first step in the system and is done at online phase. The hierarchical repository is available to all of the users. Users need to login to the system after registration. During registration personal details of the user are entered. User Preferences are specified at the online phase.ie: Personal Interests of the user. User can specify the sensitive topics in the hierarchical user profile which can't be disclosed to the server [15]. User profile H, as a hierarchical representation of user interests, is a rooted subtree of R. Given two trees S and T , S is a rooted subtree of T if S can be generated from T by removing a node set X from T .It is represented as rsbtr(X; T). Each topic t ∈ H is labeled with a user support, denoted by $\sup_H$ (t) or Pr(t), which describes the users preference on the respective topic. The user support can be recursively aggregated from those specified on the leaf topics as per equation (1):

$$\sup H(t) = \sum_{t' \in C(t,H)} \sup(t) \qquad (1)$$

Customized privacy requirements can be specified with a number of sensitive-nodes (topics) in the user profile, whose disclosure (to the server) introduces privacy risk to the user. Given a user profile H, the sensitive nodes are a set of user specified sensitive topics S ⊂ H, whose subtrees are non-overlapping, i.e., $\forall s1,\ s2 \in S(s_1 \neq s_2), s_2 \nexists Subtr(S_1,H)$. The nodes with *Pr(t)* greater than a preference threshold ∝ is most preferred by the user in his profile.

With the privacy Requirement Customization, sensitivity specified by the user is used to find the cost of each node in the profile. This cost [15] of each node is used to calculate the privacy risk of the topics in the generalized profile.

Cost of each node is finding with the following steps and as per equations (2) and (3):

For each sensitive-node, *cost(t)= sen(t);*

- For each non-sensitive leaf node, *cost(t)=0;*
- For each non-sensitive internal node, *cost(t)* is :

$$Cost(t) = \sum_{t' \in C(t,H)} \cos t(t') * pr(t'|t) \qquad (2)$$

Where, $pr(t'|t) = pr(t') \div pr(t) \qquad (3)$

To address the problem with forbidding [15], a technique is proposed in the system, which detects and removes a set of nodes X from H, such that the privacy risk introduced by exposing *G=rsbtr(X,H)* is always under control. Set X is typically different from S assuming that all the subtrees of H rooted at the nodes in X do not overlap each other. This process is called generalization, and the output G is a generalized profile.

A more flexible solution requires online generalization, which depends on the queries. Online generalization not only avoids unnecessary privacy disclosure. It randomly deletes nodes other than the most sensitive ones at runtime from user profile. At each time of query execution different generalized profiles is created at runtime which can prevent attacks. By this information loss can occur, to minimize this MinimizeIL algorithm is proposed in the system. This also keeps privacy risk under control.

With the online generalization, random nodes are removed instead of forbidding method can cause information loss. To minimize the information loss caused by this, MinimizeIL algorithm is proposed in the system. It reduces the loss of information which minimizes the privacy risk. With the removal of sensitive nodes, discriminating power (DP) of the generalized profile increases.

5. MINIMIZE INFORMATION LOSS ALGORITHM

MinimizeIL algorithm tries to minimize privacy risk ie: to kept privacy risk while the creation of generalized profile, under control. The privacy risk is minimized so that it is below privacy threshold µ, which is initialized as 0.5. Another input of the algorithm is preference threshold ∝, if the preference is greater than this threshold means the user has high preference on the topic and so the topic can be removed at the time of online generalization. If the server has more documents with some topics, removal of these nodes also causes information loss. So the algorithm is implemented like a queue from the leaf topics until it reaches the root. It continuously sends each topic to the server if information loss is less, so that the privacy risk is minimized.

The privacy risk [15] when exposing G is defined as the total sensitivity contained in it, given in normalized form. In the worst case, the original profile is exposed, and the normalized risk of exposing all sensitive nodes reaches its maximum, namely 1. However, if a sensitive node is pruned and its ancestor nodes are retained during the generalization, the risk of exposing the ancestors can be evaluated. This can be done using the cost layer values computed offline. Given a generalized profile G, the unnormalized risk [15] of exposing it is recursively given by *Risk(t,g)* as per equations (4) and (5):

$$Cost(t) \text{ ; if } t \text{ is a leaf} \qquad (4)$$

$$\max\left(Cost(t), \sum_{t' \in C(t,g)} Risk(t',g) \right) \qquad (5)$$

Then, the normalized risk [15] can be obtained by dividing the un-normalized risk of the root node with the total sensitivity in H, namely *risk (t,g)*. *risk (t,g)* of generalized profile is always in the interval [0,1].

$$risk(t,g) = risk(t,g) \div \sum_{s \in S} sen(s) \qquad (6)$$

MinimizeIL algorithm also traces the current location of a user using json command from the network service provider. This location is also sends with the query and generalized profile to the server. Thus the search can provide results based on location of the user. This also can increase the search efficiency of the PWS system. For example if user searches for café, along with the personalized results location based results for the café will also get. The friends can share their information that is not sensitive. If the query of a user matches with keyword in the search history of his friends, then the results can get offline which may significantly reduces the execution time of query. Friends of a user may have similar interests, for example if they are doing research on same topic then they can share the documents those are not private. Thus the friends can access results without any search; the time taken is only for viewing the further details of the document. This depends on the speed of the network service provider.

```
Algorithm : MinimizeIL ( H,∝,µ )
Input : Personal Profile H, Privacy Threshold µ, Preference Threshold ∝
Output : Generalized profile g satisfying minimal privacy risk and server results
While $t=root(H)$ do
  if $Risk (t,H) > \mu$ then
    Remove the node $t$ from $H$
  else if $t$ has siblings having $Pr > \propto$ and No: of documents is more then
    No operation on t's siblings
  else
    Remove the sibling of t
  $t = par(t,H)$
end
Update $Risk (t,g)$
Access the location of user with the network service provider and send it along with g to the server.
if the Previous viewed results or keyword searched by any friends matches with query of the user, then
  Get the results offline.
Return g and search results
```

6. CREATION OF USER GROUPS

User can create friends ie: a community. Each user in the community has their privacy and is implemented using generalization method. They can share what they want to share and if they search for query which is previously searched by any other friend in their community, then they can also search online. Without connecting to the network the results viewed by the friends can be available for repeated queries in the community which is shown in figure 2. If they look for the similar results, then execution time of the query can be significantly reduced. Time is only taken for viewing the documents further after connecting to the server. Result retrieval time is reduced considerably. This can increase the search efficiency of the PWS search engine.

As shown in figure 2, a user can add as much of friends he/she wants. The user can view the search results which is privacy protected by the generalization method. At offline stage too, users can get results by this.

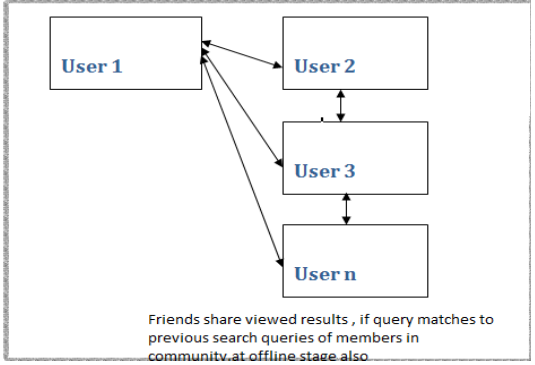
User 1
User 2
User 3
User n
Friends share viewed results , if query matches to previous search queries of members in community at offline stage also

Figure 2: Friend’s Communication

The time is taken only to link the relevant documents with the server, that is time needed to search can be avoided with the community creation. This is due to the fact that the number of results relevant to the user can be getting at an offline stage.

7. EXPERIMENTAL RESULTS

The synthetic dataset created named personal search contains mainly the data related with taxonomy repository. The relation repository contains 904 tuples indicates the human interests from which the profile is created.

Search efficiency analysis for the three different search approaches is shown in figure 3. Each recorded value of the average time taken for the result retrieval represents the maximum time limit for the corresponding operation. Each recorded value of the relevant results number represents the maximum results those are relevant to the corresponding user. The graph represents the values for retrieval of all the results related to a query. The results are returned as a list of documents and their few details of the content associated with the corresponding documents related with the query.

The noise with these search techniques can also be found out. It is calculated by dividing the number of irrelevant documents by the total number of documents gets as search results .The noise is very less in PWS with the community search system. The search efficiency (Search-Eff) is calculated as in opposite to the noise calculation as per equation (7).

$$\text{Search Eff} = \frac{\text{No:of Relevant documents}}{\text{Total No:of documents}} * 100 \qquad (7)$$

In case of normal search and PWS search, average time values for result retrieval are almost same, for majority of the inputs. Since both the results are retrieved from the same google server, the execution time is almost the same. The time taken is only for the result retrieval from server which depends on the efficiency of the network service provider. But the no: of results that are relevant to the query out of the total no: of results is very less for a normal search compared to PWS search. Normal search provides a no: of results related to the query. Sometimes this can provide more relevant results, and at sometimes may not. PWS search provides results based on the user interests, which can be more relevant to the user. Online generalization done with the minimize IL algorithm can reduce the information loss. The algorithm tries to minimize the privacy risk. Since the PWS search also take almost same time for the result retrieval, it seems to be more efficient than the normal one.

The values recorded for PWS with community search shows that the time is reduced to half for the execution of same queries other than the normal and PWS search. The time is taken only to link the relevant documents with the server, that is time needed to search can be avoided with the community creation. This is due to the fact that the number of results relevant to the user can be getting at an offline stage. The graph shows that the no: of results retrieved is same at online phase with the PWS server, but more than half results can be obtained at offline phase. The average time taken for result retrieval is reduced to half which saves time considerably. Time taken for result retrieval on both offline and online phases are recorded. Total time gets reduced to half. This increases the search efficiency of the PWS search system with community.

Figure 3 plots the queries made by a particular user against search efficiency in percentage. The efficiency for normal and PWS shows a notable difference. The search efficiency of PWS search with community is considerably more than normal PWS, which sometimes shows complete search efficiency since the average time for retrieving result is reduced to half.

Figure 4 plots the queries made by a particular user against average time of result retrieval in seconds. The time of execution for normal and PWS shows no difference. The total time taken for result retrieval of PWS search with community is reduced to half other than normal and PWS search, which sometimes give complete search efficiency. The time values also depend on various other factors such as the input query of different users, friends interests, limitation of the google services etc. Hence evaluation considering another set of input queries may result in different values. So these results cannot be considered as final.

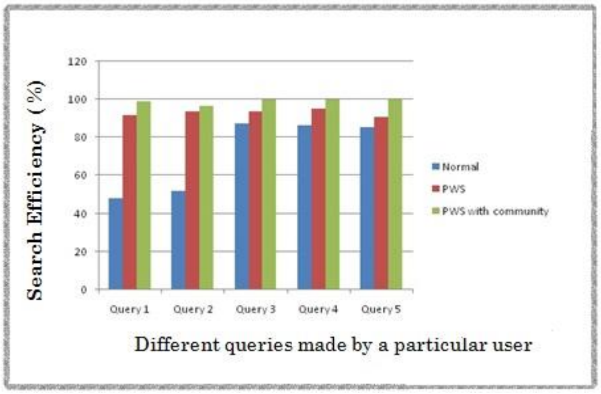

Figure 3: Search efficiency analysis

It is clear from the test results that the system returns the required results within a reasonable time limit. Normal search and PWS search with and without creation of community functionalities work without causing a notable time delay.

With this PWS system, the Privacy protection is done with creation of generalized profile. On creation of such a profile at online phase, for minimization of information loss minimize IL algorithm is used.

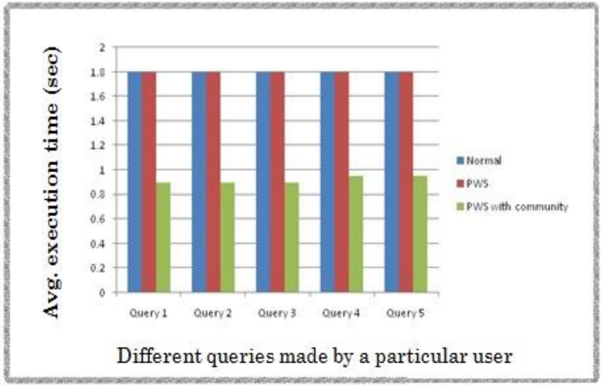

Figure 4: Execution time analysis

The privacy risk of the sensitive topics in the generalized profile is calculated for the PWS search for different queries by a particular user and the normalized risk is also finding. Values are shown in table 1. Each value of the privacy risk for the generalized profile is finding with the algorithm theoretically. The algorithm tries to minimize the privacy risk below privacy threshold µ.Normalized value will be in the range (0,1]. The table shows that the system keeps the privacy risk under control and so information loss is also get reduced.

Table 1: Privacy risk calculation

Query	Privacy risk	Normalized risk
Dance	0.1	0.06
Diabetes	0.3	0.2
Animation	0.1	0.06
Music	0.2	0.13
Computer	0.4	0.26

Figure 5 plots the privacy risk and normalized risk of generalized profile created on different queries of a particular user against privacy threshold. The graph shows that privacy risk is kept under control while creation of generalized profiles.

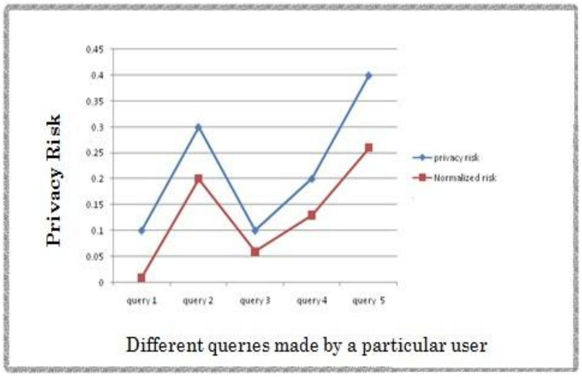

Figure 5: Privacy risk analysis

8. CONCLUSION AND FUTURE WORK

Personalization is an attempt to find most relevant documents using information about user's goals, knowledge, preferences etc. Since with the growth of data links in a search engine, even a good query can return not just ten's, but thousand's of relevant documents. 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. This raises privacy issues which prevents wide proliferation of PWS. The proposed system introduces an enhanced generalization method at online phase, which keeps the privacy risk under control. The time of execution depends upon the network speed and server performance. This time can be reduced with the creation of a community in this system. This significantly reduces the execution time of query, by retrieving results at an offline stage. All the essential requirements for PWS are met with minimum overhead.

The proposed system relies on a dataset for profile creation. It includes only limited information and hence it may not be able to cover all the human preferences in the entire human knowledge. The community members may also search for adhoc queries which are not at all related to the interests of other users. The services provided by ajax google services is also limited. The system efficiency can be improved if noisy topics in the profile are avoided along with runtime generalization. This can increase the no: of relevant results or can completely avoid noise. The quality of retrieved results can be further improved if the query processing module can infer the actual intention of the user from the query. These enhancements incur additional processing overhead, but it is expected to improve the performance of the system.

References

[1] 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.

[2] J. Teevan, S.T. Dumais, and D.J. Liebling, "To Personalize or Not to Personalize: Modeling Queries with Variation in User Intent," Proc. 31st Ann. Intl ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR),pp. 163-170,2008.

[3] 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.

[4] H.H. Crokell, K. Hafner, "Researchers Yearn to Use AOL Logs, but They Hesitate," New York Times, Aug. 2006.

[5] 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.

[6] 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.

[7] Y. Xu, K. Wang, G. Yang, and A.W.C Fu, "Online anonymity for personalized web services," Proc.18th ACM conformation and knowledge management (CIKM), pp1497-1500, 2009.

[8] J. Castelli-Roca and A. Vijeo and J. Herrera-Joancomarti, "Preserving users privacy in web search engines," Computer Comm. Vol.32, pp 1541-1551, 2009.

[9] Viejo and J. Castella-Roca, "Using Social Networks to Distort Users Profiles Generated by Web Search Engines," Computer Networks, vol. 54, pp. 1343-1357, 2010.

[10] A. Krause and E. Horvitz, "A Utility-Theoretic Approach to Privacy in Online Services," J. Artificial Intelligence Research ,vol. 39, pp. 633-662, 2010.

[11] X. Shen, B. Tan, and C. Zhai, "Privacy Protection in Personalized Search," ACM SIGIR Forum, vol. 41, pp. 4-17, 2007.

[12] X. Xiao and Y. Tao, "Personalized Privacy Preservation," Proc. ACM SIGMOD Intl Conf. Management of Data (SIGMOD), 2006.

[13] M. Spertta and S. Gach, "Personalizing Search Based on User Search Histories," Proc. IEEE/ WIC/ACM Intl Conf. Web Intelligence (WI), 2005.

[14] G. Chen, H. Bai, L. Shou, K. Chen, and Y. Ga, "Ups: Efficient Privacy Protection in Personalized Web Search," Proc. 34th Intl ACM SIGIR Conf. Research and Development in Information, pp. 615-624, 2011.

[15] Lidan Shou, He Bai, Ke Chen, and Gang Chen , "Supporting privacy protection in personalized web search," IEEE Transactions on knowledge and data engineering, Vol. 26, No.2, February 2014.

[16] Y. Zhu, L. Xiong and C. Verdery , "Anonymizing user profiles for personalized web search," Proc.19th Int'l conf. World Wide Web(WWW) , pp.1225-1226, 2010.

[17] 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, 2007.