Implementation of Personalized Web Search with Efficient Privacy Protection

[1] **Veernala Neeharika**

[1]M.Tech Scholar, Department of Computer Science and System Engineering,

Andhra University College of Engineering (A), Visakhapatnam, AP, India.

[2]Department of Computer Science and System Engineering,

Andhra University College of Engineering (A), Visakhapatnam, AP, India.

Abstract: Internet of things is in its mount in today’s world. Web exploring is the most common task performs on the internet. The web search engines are the most important tool of the internet; search engines are the place from where an individual can collect the relevant information and search according to keyword given by the user. The data on the wed are increasing day by day very dramatically. The user has to spend a lot of time on the net for finding the data in which they are interested. The large-scale user-generated meta-data not only facilitate users in sharing and organizing multimedia content, but provide useful information to improve media retrieval and management. Personalized search serves as one of such examples where the web search experience is improved by generating the returned list according to the modified user search intents. In this paper, we exploit the annotations and propose a novel framework simultaneously considering the user and query relevance to learn to personalized content like that image search. For minimizing the privacy risk here we propose the client side based technique with the combination of Greedy method to prevent the user data that we applied in Knowledge mining area. The PWS techniques mainly depends on the contents of web mining, browsing information, links, individual user profile and also queries. The proposed paper is to study on different strategies of personalization. PWS framework called UPS can adaptively generalize profiles by queries while respecting user specified privacy requirements. 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. Two greedy algorithms, namely Greedydp and GreedyIL, are used for runtime generalization. An online prediction mechanism for deciding whether personalizing a query is beneficial is provided. Extensive experiments demonstrate the effectiveness of the framework. The experimental results also reveal that GreedyIL significantly outperforms GreedyDP in terms of efficiency Privacy protection in PWS applications can be adopted that model user preferences as hierarchical user profiles by studying a PWS framework called UPS that adaptively generalizes profiles by queries while keeping in mind user-specified privacy requirements..

***Index Terms* – Web Search Engine, personalized search, user query, content search and privacy preserving, Privacy risk, Profile.**

I. Introduction

Internet of things is in its mount in today’s world. Web exploring is the most common task performs on the internet. The web search engines are the most important tool of the internet; search engines are the place from where an individual can collect the relevant information and search according to keyword given by the user. The data is increasing day by day very dramatically. The user has to spend a lot of time on the net for finding the data in which they are interested. The irrelevant result may irritate the user and hence, the efficiency of the query search should be improved. To improve the search, personalized web search framework has demonstrated to retrieve the data on the interest. A great many electronic information are incorporated on many millions information that are already on-line today. Data mining is characterized as the programmed extraction of obscure, valuable and reasonable patterns from extensive database. Tremendous occurrence of web expands the complexity for all kinds of people to search effectively. To expand the execution of sites better site design, web server actions are changed according to users' interests. Web mining means the utilization of data mining concepts to consequently recover, remove and assess data for learning disclosure from web documents.

The web search engine has long become the most important portal for ordinary people looking for useful information on the web. However, users might experience failure when search engines return irrelevant results that do not meet their real intentions. Such irrelevance is largely due to the enormous variety of users’ contexts and backgrounds, as well as the ambiguity of texts. 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 analysed to figure out the user intention behind the issued query. The solutions to PWS can generally be categorized into two types, namely click-log-based methods and profile-based ones. The click-log based methods are straightforward they simply impose bias to clicked pages in the user’s query history. Although this strategy has been demonstrated to perform consistently and considerably well it can only work on repeated queries from the same user, which is a strong limitation confining its applicability. In contrast, profile-based methods improve the search experience with complicated user-interest models generated from user profiling techniques. Profile-based methods can be potentially effective for almost all sorts of queries, but are reported to be unstable under some circumstances. Although there are pros and cons for both types of PWS techniques, the profile-based PWS has demonstrated more effectiveness in improving the quality of web search recently, with increasing usage of personal and behaviour information to profile its users, which is usually gathered implicitly from query history, browsing history click-through data bookmarks, user documents, and so forth.

The Personalized Web Search provides a unique opportunity to consolidate and scrutinize the work from industrial labs on personalizing web search using user logged search behaviour context. It presents a fully anonymized dataset, which has anonymized user id, queries based on the keywords, their terms of query, providing URLs, domain of URL and the user clicks. This dispute and the shared dataset will enable a whole new set of researchers to study the problem of personalizing web search experience. It decreases the likelihood of finding new information by biasing search results towards what the user has already found. By using these methods privacy of the user might be loss because of clicking the relevant search, frequently visited sites and providing their personal information like their name, address, etc. in this case their privacy might be leak. For this privacy issue, many existing work proposed a potential privacy problems in which a user may not be aware that their search results are personalized for them [6, 7]. Unfortunately, such implicitly collected personal data can easily reveal a gamut of user's private life. Privacy issues rising from the lack of protection for such data, for instance the AOL query logs scandal, not only raise panic among individual users, but also dampen the data-publisher's enthusiasm in offering personalized service. In fact, privacy concerns have become the major barrier for wide proliferation of PWS services. To protect user privacy in profile-based PWS, researchers have to consider two contradicting effects during the search process. On the one hand, they attempt to improve the search quality with the personalization utility of the user profile. On the other hand, they need to hide the privacy contents existing in the user profile to place the privacy risk under control.

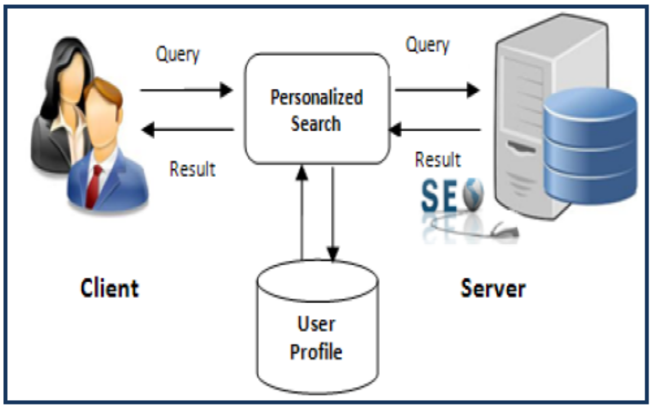

Fig 1: Personalized Search Engine Architecture

The generalization process is guided by considering two conflicting metrics, namely 1. the personalization utility and the privacy risk, both defined for user profiles. 2. Subsequently, the query and the generalized user profile are sent together to the PWS server for personalized search. 3. The search results are personalized with the profile and delivered back to the query proxy. 4. Finally, the proxy either presents the raw results to the user, or re-ranks them with the complete user profile. UPS is distinguished from conventional PWS in that it 1) provides runtime profiling, which in effect optimizes the personalization utility while respecting user's privacy requirements; 2) allows for customization of privacy needs; and 3) does not require iterative user interaction. Our main contributions are summarized as following: We propose a privacy-preserving personalized web search framework UPS, which can generalize profiles for each query according to user-specified privacy requirements. Relying on the definition of two conflicting metrics, namely personalization utility and privacy risk, for hierarchical user profile, we formulate the problem of privacy-preserving personalized search as Risk Profile Generalization, with its NP-hardness proved. We develop two 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 provide 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.

III .Related Work

In [1] Z. Dou, R. Song, and J.-R. Wen et al. Personalization strategies had been proposed and investigated for many years but it's miles nonetheless doubtful whether or not the strategy is always effective on distinctive queries for special customers, under one of a kind search context. In [1], they have investigated whether personalization is continuously effective below distinctive conditions. They advanced an evaluation framework based totally on question logs to allow big scale assessment of personalized search. Click entropy an easy size on whether the question must be personalized. Click primarily based personalization strategies can paintings on repeated queries. The benefits revealed that the personalization has different effectiveness on different queries and both short term and long term context improve the search performance. On the other side, because of a large-scale evaluation of search contexts, the framework may be time-consuming and complex to handle.

In [2] A. Krause and E. Horvitz et al. Online offerings, for example, web search, news portals, and e-commerce applications confront the test of giving amazing support of an expansive, heterogeneous client base. To overcome such problem an effort has been introduced by introducing methods to personalize services based on special knowledge about users and their context. Researchers and organizations have sought after explicit and implicit methods for customizing online administrations. An approach for explicitly optimizing the utility-privacy trade off in personalized services such as web search. Privacy concerns show super-modularity; the more private information we accrue, the faster sensitivity and the risk of identifiability grow.

[2]A. Krause et al demonstrated how can efficiently find a provably near- optimal utility-privacy tradeoff and evaluated methodology on real-world web search data. The common belief is that the principles and methods employed in the utility-theoretic analysis of tradeoffs for web search have applicability to the personalization of a broad variety of online services. In [2] found that significant personalization can be achieved using only a small amount of information about users with the limitation that the system is dependent on the log of user search activity.

In [3] J. Castelli-Roca, A. Viejo, and J. Herrera-Joancomarti et al. Web search engines like Yahoo!, Google, Bing, etc. are widely used to find the particular amount of data among a large amount of data in a short amount of time. People over the globe use the web search engine for different purposes which are relevant to them. At the same time, needed information belongs to the specific topic is hidden among all the available data and it can be really difficult to find it since that information can be separated all over the World Wide Web. In fact, these useful things can also cause the privacy threats to the users, web search engines can profile the client by storing and analyzing the past queries requested by them. But to solve this privacy threats current mechanism introduces high cost in terms of computation and communication. In this paper, they produce a novel protocol designed to protect the user's privacy in front of web search profiling. Their system gives the duplicate or deformed user profile to the web search engines. [3] They offered implementation details and computational or communication results which show that the introduced protocol improves the existing solutions in terms of query delay. The limitation of the existing system was that the person or the entity can get some advantage over the other benefits from the absence of privacy protection mechanism between the user and the web search engine. So the problem of submitting the queries of the user to the search engine while preserving the privacy protection to the profile it can be term as Private Information Retrieval (PIR) problem. In PIR what happen is user can retrieve his values from the database while the server gets no information about the activity of the user. Simple methods to obtain the certain level of privacy to the web browsing includes the use of the proxies or the dynamic IP address. But proxy does not solve the privacy problem. The proxy can prevent the web search engines from creating the profile of the user, it can profile them instead.

In [4] X. Xiao and Y. Tao et al. did study on the generalization for preserving the privacy of the sensitive data which is daily produced by the users. The existing techniques concentrate on the each and every approach that cause the same amount of preservation for all the users without analyzing their original needs. This results in providing the insufficient protection to a group of people who actually need it while giving extreme privacy control to the group of people who doesn't need it. This system cannot guarantee the privacy protection in all cases this could lead to cause the unnecessary data loss by performing excessive use of generalization. At first, they make a concept that forms a new framework of computing privacy which takes into account the sensible information by an individual preference. Secondly, they analyze the theory behind their methodology and evaluate the formulae for quantifying the privacy which clearly show the scenarios where k-anonymity may make sure about safe data production. Finally, they evolved an algorithm for finding the generalized that keeps a huge amount of information in the microdata without breaking any privacy limits. The Greedy Algorithm divided into two categories, according to the constraint imposed on generalization. The first category includes "full-domain generalization" which undertake hierarchy on every QI attribute and all the partitions in the hierarchy needs to be at same level. The second category includes "full-sub tree recording" which drop the same level of hierarchy which mentioned earlier in the first category that causes unnecessary information lose.

In this section, the related works are overviewed. Focus is on the literature of profile-based personalization and privacy protection in PWS system.

A. Profile-based personalization there has been several prior attempts to personalize Web search. One approach to personalization is to have users describe their general interests. For example, Google Personal asks users to build a profile of them by selecting categories of interests. This profile can then be used to personalize search results by mapping Web pages to the same categories. Many commercial information filtering systems use this approach, and it has been explored before to personalize Web search results. Personal profiles have also been used in the context of the Web search to create a personalized version of PageRank [10] for setting the query-independent priors on Web pages. A similar technique for mapping user queries to categories based on the user's search history. Actually, this framework can potentially adopt any hierarchical representation based on a taxonomy of knowledge. As for the performance measures of PWS in the literature, Normalized Discounted Cumulative Gain (nDCG) is a common measure of the effectiveness of an information retrieval system. It is based on a human-graded relevance scale of item-positions in the result list, and is, therefore, known for its high cost in explicit feedback collection. To reduce the human involvement in performance measuring, researchers also propose other metrics of personalized web search that rely on clicking decisions, including Average Precision, Rank Scoring and Average Rank [3]. Average Precision metric, proposed by Dou et al. [1], to measure the effectiveness of the personalization in UPS. Meanwhile, our work is distinguished from previous studies as it also proposes two predictive metrics, namely personalization utility and privacy risk, on a profile instance without requesting for user feedback

B. Privacy Protection in PWS System Typical works in the literature of protecting user identifications try to solve the privacy problem on different levels, including the pseudo-identity, the group identity, no identity, and no personal information. Solution to the first level is proved to 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. Both [8] and [9] provide online anonymity 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. The useless user profile (UUP) protocol is proposed 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. A more important property that distinguishes our work from [10] is that we provide personalized privacy protection in PWS. A person can specify the degree of privacy protection for her/his sensitive values by specifying "guarding nodes" in the taxonomy of the sensitive attribute. Motivate by this, we allow users to customize privacy needs in their hierarchical user profiles. Aside from the above works, a couple of recent studies have raised an interesting question that concerns the privacy protection in PWS.

C. Slicing Two popular Anonymization techniques are generalization and bucketization. Generalization, replaces a value with a "less-specific but semantically consistent" value. The main problems with generalization are:

- It fails on high-dimensional data due to the curse of dimensionality.
- It causes too much information loss due to the uniform-distribution assumption.

Bucketization first partitions tuples in the table into buckets and then separates the quasi identifiers with the sensitive attribute by randomly permuting the sensitive attribute values in each bucket. The anonymized data consist of a set of buckets with permuted sensitive attribute values. In particular, bucketization has been used for anonymizing high-dimensional data. However, their approach assumes a clear separation between Qis and SAs. In addition, because the exact values of all QIs are released, membership information is disclosed. The key idea of slicing is to preserve correlations between highly correlated attributes and to break correlations between uncorrelated attributes thus achieving both better utility and better privacy. Third, existing data analysis (e.g. query answering) methods can be easily used on the sliced data.

IV. PROBLEM DEFINITION

Most of the existing works concentrate on server-side personalized search services in preserving privacy, it provide a less security to the user. To provide a security to the user from the profile-based PWS from the client side, many researchers have to deem two challenging effects during the search process of the user, (i) To increase the search quality by user profile and (ii) hide the privacy content to place the privacy risk under control. In many studies tells that user suggestions and their click based method is the helpful way to provide a personalized search and at the same time they have trouble with the loss of their privacy under their providing contents. Profile based method is an ideal case for providing the relevant search [18, 19]. Under this they were many drawbacks, it does not support on the runtime profiling, it can be based on the online and offline generalization, insufficiently protection of the data and require more iteration for obtaining relevant search.

The issue with the existing method are explained in following remarks:

1. Profile-based Personalized Web Search has a disadvantage that it do not support runtime profiling. A user profile is typically generalized for only once offline and it may not even improve the search quality for some ad hoc queries, exposing user profile to a server has put the user's privacy at risk.

2. The existing methods do not take into account the customization of privacy requirements. This probably makes some user privacy to be overprotected while others insufficiently protected.

3. Most of the personalization techniques need repetition of user interaction when building up the personalized search results. The result with some metric which require multiple user interactions like rank scoring, average rank [8], and so on.

V. PROPOSED SYSTEM

Indeed, the privacy concern is one of the major barriers in deploying serious personalized search applications, and how to attain personalized search though preserving users' privacy. Here we propose a client side personalization which deals with the preserving privacy and envision possible future strategies to fully protect user privacy. For privacy, we introduce our approach to digitalized multimedia content based on user profile information. For this, two main methods were developed: Automatic creation of user profiles based on our profile generator mechanism and on the other hand recommendation system based on the content to estimates the user interest based on our client side meta data.

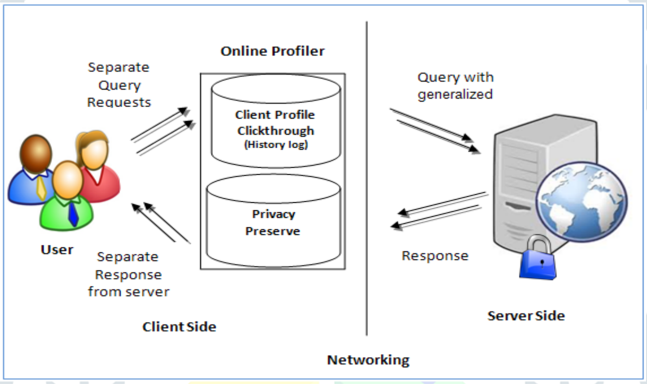

Fig 2: Proposed Architecture

Above figure shows our proposed architecture which is builds in the client side mechanism and here we protect the data from the server, so only we provides a privacy to the client user. Every query from the client user were provided by the separate requests to the server, this hides the frequent click through logs or content based mechanism, from this user can protect the data from the server. In the same case our mechanism maintains the online profiler about the user hence it hides the click logs and provides a safeguard to the user data. After that, online profiler query were processed in the manner of generalization process, it is used to meet the specific prerequisites to handle the user profile and it is based on the preprocessing the user profiles. Our architecture, not only the user's search performance but also their background activities (e.g., viewed before) and personal information (e.g., emails, browser bookmarks) could be included into the user profile, permitting for the structure of a much richer user model for personalization. The sensitive contextual information is usually not a main aspect since it is strictly stored and used on the client side. A user's personal information including user queries and click logs history resides on the user's personal computer, and is exploited to better suppose the user' information require and provide a relevant search results. Our proposed algorithm uses the greedy method based on the discriminating power and information loss protection to inherit the relations. Here it uses the inherited method to generalize the query. It allows performing the customization process to protect the data and use the User customizable Privacy-preserving Search framework addressed the privacy problems. This aims at protecting the privacy in individual user profiles.

METHODOLOGY

This project has carried by various methods and algorithms all these are implemented to obtain the desired result. As there are certain methodologies in used in this project and are explained below.

Profile-Based Personalization

This paper introduces an approach to personalize digital multimedia content based on user profile information. For this, two main mechanisms were developed: a profile generator that automatically creates user profiles representing the user preferences, and a content-based recommendation algorithm that estimates the user's interest in unknown content by matching her profile to metadata descriptions of the content. Both features are integrated into a personalization system.

Privacy Protection in PWS System

We propose a PWS framework called UPS that can generalize profiles in for each query according to user-specified privacy requirements. Two predictive metrics are proposed to evaluate the privacy breach risk and the query utility for hierarchical user profile. We develop two simple but effective generalization algorithms for user profiles allowing for query-level customization using our proposed metrics. We also provide an online prediction mechanism based on query utility for deciding whether to personalize a query in UPS. Extensive experiments demonstrate the efficiency and effectiveness of our framework. Therefore, the need for personalization becomes questionable for such queries. While these works are motivated in questioning whether to personalize or not to, they assume the availability of massive user query logs (on the server side) and user feedback. In our UPS framework, we differentiate distinct queries from ambiguous ones based on a client-side solution using the predictive query utility metric

Generalizing User Profile

The generalization process has to meet specific prerequisites to handle the user profile. This is achieved by preprocessing the user profile. At first, the process initializes the user profile by taking the indicated parent user profile into account. The process adds the inherited properties to the properties of the local user profile. Thereafter the process loads the data for the foreground and the background of the map according to the described selection in the user profile.

Additionally, using references enables caching and is helpful when considering an implementation in a production environment. The reference to the user profile can be used as an identifier for already processed user profiles. It allows performing the customization process once, but reusing the result multiple times. However, it has to be made sure, that an update of the user profile is also propagated to the generalization process. This requires specific update strategies, which check after a specific timeout or a specific event, if the user profile has not changed yet. Additionally, as the generalization process involves remote data services, which might be updated frequently, the cached generalization results might become outdated. Thus selecting a specific caching strategy requires careful analysis.

Attack Model

This represents the chance of attackers attacking upon the data. The search query can be captured by the intruder and can make use of the information. This is like eavesdropping;

Knowledge bounded: The background knowledge of the adversary is limited to the taxonomy repository R. Both the profile H and privacy are defined based on R Session bounded: None of previously captured information is available for tracing the same victim in a long duration. In other words, the eavesdropping will be started and ended within a single query session.

Online Decision

The profile-based personalization contributes little or even reduces the search quality, while exposing the profile to a server would for sure risk the user's privacy. To address this problem, we develop an online mechanism to decide whether to personalize a query. The basic idea is straightforward. if a distinct query is identified during generalization, the entire runtime profiling will be aborted and the query will be sent to the server without a user profile.

SLICING ALGORITHM

Many algorithms like bucketization, generalization have tried to preserve privacy however they exhibit attribute disclosure. So to overcome this problem an algorithm called slicing is used. This algorithm consists of three phases: attribute partitioning, column generalization, and tuple partitioning.

A. Attribute Partitioning

This algorithm partitions attributes so that highly correlated attributes are in the same column. This is good for both utility and privacy. In terms of data utility, grouping highly correlated attributes preserves the correlations among those attributes. In terms of privacy, the association of uncorrelated attributes presents higher identification risks than the association of highly correlated

attributes because the associations of uncorrelated attribute values is much less frequent and thus more identifiable.

B. Column Generalization

Although column generalization is not a required phase, it can be useful in several aspects. First, column generalization may be required for identity/membership disclosure protection. If a column value is unique in a column (i.e., the column value appears only once in the column), a tuple with this unique column value can only have one matching bucket. The main problem is that this unique column value can be identifying. In this case, it would be useful to apply column generalization to ensure that each column value appears with at least some frequency. Second, when column generalization is applied, to achieve the same level of privacy against attribute disclosure, bucket sizes can be smaller. While column generalization may result in information loss, smaller bucket-sizes allow better data utility. Therefore, there is a trade-off between column generalization and tuple partitioning.

C. Tuple Partitioning

The algorithm maintains two data structures: 1) a queue of buckets Q and 2) a set of sliced buckets SB. Initially, Q contains only one bucket which includes all tuples and SB is empty. For each iteration, the algorithm removes a bucket from Q and splits the bucket into two buckets. If the sliced table after the split satisfies l-diversity, then the algorithm puts the two buckets at the end of the queue Q Otherwise, we cannot split the bucket anymore and the algorithm puts the bucket into SB. When Q becomes empty, we have computed the sliced table. The set of sliced buckets is SB

GENERALIZATION ALGORITHMS

A. The GreedyDP Algorithm

Given the complexity of our problem, a more practical solution would be a near-optimal greedy algorithm. As preliminary, we introduce an operator -t called prune-leaf, which indicates the removal of a leaf topic t from a profile. Formally, we denote by Gi -t Gi+1 the process of pruning leaf t from Gi to obtain Gi+1. Obviously, the optimal profile G0 can be generated with a finite-length transitive closure of prune-leaf. The first greedy algorithm GreedyDP works in a bottom up manner. Starting from G0, in every ith iteration, GreedyDP chooses a leaf topic t ε TGi (q) for pruning, trying to maximize the utility of the output of the current iteration, namely Gi+1. During the iterations, we also maintain a best -profile- so-far, which indicates the Gi+1 having the highest discriminating power while satisfying the -risk constraint. The iterative process terminates when the profile is generalized to a root-topic. The best-profile so far will be the final result (G*) of the algorithm. The main problem of GreedyDP is that it requires recomputation of all candidate profiles (together with their discriminating power and privacy risk) generated from attempts of pruneleaf on all t ε TGi(q). This causes significant memory requirements and computational cost.

B. The GreedyIL algorithm

It improves the efficiency of the generalization using heuristics based on several findings. One important finding is that any prune-leaf operation reduces the discriminating power of the profile. In other words, the DP displays monotonicity by prune-leaf. Three following heuristics extends this algorithm:

- The iterative process can terminate whenever δ-risk is satisfied.
- Once a leaf topic t is pruned, only the candidate operators pruning t's sibling topics need to be updated in Q.

CONCLUSION AND FUTURE WORK

The search history and the search queries of the web user are saved by the web search engines. This saved data can be used by the user as to provide other relevant data for the user. User personal data i.e. browsing histories and the queries create the profile of the user by the engines and it should be protected to avoid the threats. UPS could be used by any typical PWS that takes users profiles in a hierarchical structure. The generalization algorithms, GreedyDP, and IL, which handles the privacy issues in PWS by offering user to control the amount of private data reveal to the web servers. The private parameters facilitate smooth control of privacy exposure while maintaining good ranking quality. In future, other privacy threats can be handled with efficient algorithm and can find smarter techniques to build the user profile, and better metrics to predict the performance of UPS. We performed some experiments that shows better search result when we use advanced user profile as compared with simple user profile on same queries. In the future we would try to enhance the search quality based on user search preference and also aim to provide more security from the adversaries. For this issue this paper proposes client based architecture based on the greedy algorithm to prevent the user data and provide the relevant search result to the user in future it can include this work in mobile application.

REFERENCES

[1]. 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.

[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]. 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.

[5]. 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.

[6]. 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.

[7]. X. Shen, B. Tan, and C. Zhai, “Privacy Protection in Personalized Search,” SIGIR Forum, vol. 41, no. 1, pp. 4-17, 2007.

[8]. 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.

[9]. Dou, Zhicheng, Ruihua Song, and Ji-Rong Wen. "A large-scale evaluation and analysis of personalized search strategies", Proceedings of the 16th international conference on World Wide Web. ACM, 2007.

[10]. 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. Int’l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR), pp. 163-170, 2008.

[11]. Shen, Xuehua, Bin Tan, and Cheng Xiang Zhai. "Implicit user modeling for personalized search." Proceedings of the 14th ACM international conference on Information and knowledge management. ACM, 2005.

[12]. T. Joachims, L. Granka, B. Pang, H. Hembrooke, and G. Gay, “Accurately Interpreting Clickthrough Data as Implicit Feedback,” Proc. 28th Ann. Int’l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR ’05), pp. 154-161, 2005.

[13]. Shen, Xuehua, Bin Tan, and Cheng Xiang Zhai. "Context-sensitive information retrieval using implicit feedback." Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 2005.