ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Supportive Confidentiality security Based Personalized Web Search

M Anil Kumar
PG Scholar,
Department of CSE,
AVN Institute of Engineering & Technology.

Dr.S.K.Abdul Nabi
HOD,
Department of CSE,
AVN Institute of Engineering & Technology.

ABSTRACT:

Personalized web search is a promising way to improve search quality by customizing search results for people with individual information goals. However, users are uncomfortable with exposing private preference information to search engines. On the other hand, privacy is not absolute, and often can be compromised if there is a gain in service or profitability to the user. Thus, a balance must be struck between search quality and privacy protection. This paper presents a scalable way for users to automatically build rich user profiles. These profiles summarize a user's interests into a hierarchical organization according to specific interests. Two parameters for specifying privacy requirements are proposed to help the user to choose the content and degree of detail of the profile information that is exposed to the search engine. Experiments showed that the user profile improved search quality when compared to standard MSN rankings. More importantly, results verified our hypothesis that 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.

Keywords:
privacy, personalized search, hierarchical user profile.

1. INTRODUCTION:

As the amount of information on the web continuously grows, it has become increasingly difficult for web search engines to find information that satisfies users' individual needs. Personalized search is a promising way to improve search quality by customizing search results for people with different information goals. Many recent research efforts have focused on this area.

Most of them could be categorized into two general approaches: Re-ranking query results returned by search engines locally using personal information; or sending personal information and queries together to the search engine [1]. A good personalizationalgorithm relies on rich user profiles and web corpus. However, as the web corpus is on the server, re-ranking on the client side is bandwidth intensive because it requires a large number of search results transmitted to the client before re- ranking. Alternatively, if the amount of information transmitted is limited through filtering on the server side, it pins high hope on the existence of desired information among filtered results, which is not always the case. Therefore, most of personalized search services online like Google Personalized Search [2] and Yahoo! My Web[3] adopt the second approach to tailor results on the server by analyzing collected personal information, e.g. personal interests, and search histories.

Nonetheless, this approach has privacy issues on exposing personal information to a public server. It usually requires users to grant the server full access to their personal and behavior information on the Internet. Without the user's permission, gleaning such information would violate an individual's privacy. In particular, Canada launched the Personal InformationProtection and Electronic Document Act1in2001 to protect awide spectrum of information, i.e., age, race, income, evaluations, and even intentions to acquire goods or services from being released to outside parties. It is also evidenced by a recent survey conducted by Choicestream2 that the privacy fear continues to escalate although personalization remains something most consumers want. The number of consumers interested in personalization remains at a remarkably high 80%; however, only 32% of respondents were willing to share personal information in exchange for personalized experience,

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

down from 41% in 2004. Recent coverage about identity thefts and online security breaches, i.e. AOL search query data scandal, even causes users to be more wary than ever on sharing their private information—even with established, trusted brands.Thus, people maycompromise some personal information if this yields them some gain in service quality or profitability. Another important observation is that detailed personal information might not benecessary if it is possible to catch a user's interests at more general level. In the above example, the times and locationswhere the user has played basketball would not be relevant in searching for a favorite NBA basketball team. In fact, such unnecessarily detailed information often becomes noise in the search task. Hence, a proper filtering of a user's private information not only helps protect the user's privacy but also may help improve the search quality. The key is distinguishing between useful information and noise, as well as striking balance between search quality and privacy protection.

Personal data, i.e. browsing history, emails, etc., are mostly unstructured, for which it is hard to measure privacy. In addition, it is also difficult to incorporate unstructured data with search engines without summarization. So, for the purpose of both web personalization and privacy preservation, it is necessary for an algorithm to collect, summarize, and organize a user's personal information into a structured user profile. Meanwhile, the notion of privacy is highly subjective and depends on the individuals involved. Things considered to be private by one person could be something that others would love to share. In this regard, the user should have control over which parts of the user profile is shared with the server.This paper targets at bridging the conflict needs of personalization and privacy protection, and provides a solution where users decide their own privacy settings based on a structured user profile. This benefits the user in the following ways:

* Offers a scalable way to automatically build a Privacy concerns are natural and important especially on the hierarchical user profile on the client side. It's notrealistic to require that every user to specify their personal interests explicitly and clearly. Thus, an algorithm is implemented to automatically collect personal information that indicates an implicit goal or intent. The user profile is built hierarchically so that the higher-level interests are more general, and the lower -level interests are more specific. In this approach, a rich pool of profile sources is explored including browsing histories, emails and personal documents.

* Offers an easy way to protect and measure privacy. Witha hierarchical user profile, the exposure of private information is controlled using two parameters. minDetail determines which part of user profile is protected. Interests in the user profile that does not satisfy minDetail are either too specific or uncommon, are considered private and hidden from the server. expRatio measures how much private information is exposed or protected for a specified minDetail.

The paper is organized as follows: Section 2 reviews related work focusing on personalized search and privacy issues. An overview of the problem is given in Section 3. Our approach is described in Section 4. Experiment results are presented in Section 5. Conclusions are presented in Section 6.

2. RELATED WORK:

In information retrieval, much research is focused on personalized search. Relevance feedback and query refinement [13] [14] harnesses a short-term model of a user's interests, and information about a user's intent is collected at query time. Personalinformation has also been used in the context of Web search to create a personalized version of PageRank [5] [6]. There are still approaches, including many commercially available information-filtering systems [9] [10], which require users explicitly specify their interests. However, as [13] pointed out, users are typically unwilling to spend the extra effort on specifying their intentions. Even if they are motivated, they are not always successful in doing so.

A majority of work focuses on implicitly building user profiles to infer a user's intention. A wide range of implicit user activities have been proposed as sources of enhanced search information. This includes a user's search history [12], browsing history [7], click- through data [18] [28], web community [12] [15], and rich client side information [8] in the form of desktop indices. Our approach is open to all kinds of different data sources for building user profiles, provided the sources can be extracted into text. In our experiments data sources like IE histories, emails and recent personal documents were tested.User profiles can be represented by a weighted term vector [7], weighted concept hierarchical structures [10] [12] like ODP3, or other implicit user interest hierarchy [11]. For the purposes of selectively exposing users' interests to search engines, the user profile is a term based hierarchical structure that is related to frequent term based clustering algorithms [16][17]. The difference here is that the hierarchical structure is implicitly constructed in a top-down fashion. And the focus is the relationships among terms, not clustering the terms into groups.

Internet. Some prior studies on Private Information Retrieval (PIR) [4], focuses on the problem of allowing the user to retrieve information while keeping the query private. Instead, this study targets preserving privacy of the user profile, while still benefiting from selective access to general information that the user agrees to release. To our knowledge, this problem has not been studied in the context of personalized search. One possible reason for this is that personal information, i.e. browsing history and emails, is mostly unstructured data, for which privacy is difficult to measure and quantify.Some works on privacy issues in the data mining community focus on protecting individual data entries while allowing information summarization. A popular way of measuring privacy in data mining is by examining the difference in prior and posterior knowledge of a specific value [19] [20]. This can be formalized as the conditional probability or Shannon's information theory. Another way to measure privacy is the notion of k-anonymity [21] which advocates that personally identifying attributes be generalized such that each person is indistinguishable from at least k-1 other persons. In this study the notion of privacy does not compare information from different users, but rather the information collected over time for a single user. In addition, this study addresses unstructured data.

3. PROBLEM OVERVIEW:

Personal data, i.e. personal documents, browsing history and emails might be helpful to identify a user's implicit intents.However, users have concerns about how their personal information is used. Privacy, as opposed to security or confidentiality, highly depends on the person involved and how that person may benefit from sharing personal information. The question here is whether a solution can be found where users themselves are able to set their own privacy levels for user profiles to improve the search quality.

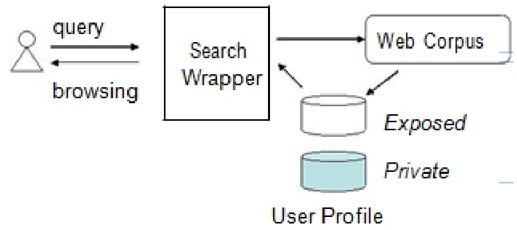

Figure 1. System Overview

Figure 1 provides an overview of the whole system. An algorithm is provided for the user to automatically build a hierarchical user profile that represents the user's implicit personal interests. General interests are put on a higher level; specific interests are put on a lower level. Only portions of the user profile will be exposed to the search engine in accordance with a user's own privacy settings. 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.The solution has three parts: First, a scalable algorithm automatically builds a hierarchical user profile from available source data. Then, privacy parameters are offered to the user to determine the content and amount of personal information that will be revealed. Third, a search engine wrapper personalizes the search results with the help of the partial user profile.

4. PRIVACY-ENHANCING: PERSONALIZED SEARCH

4.1 Constructing a Hierarchical User Profile

Any personal documents such as browsing history and emails on a user's computer could be the data source for user profiles. Our hypothesis is that terms that frequently appear in such documents represent topics that interest users. This focus on frequent terms limits the dimensionality of the document set, which further provides a clear description of users' interest. This approach proposes to build a hierarchical user profile based on frequent terms. In the hierarchy, general terms with higher frequency are placed at higher levels, and specific terms with lower frequency are placed at lower levels.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

D represents the collection of all personal documents and eachdocument is treated as a list of terms. D(t) denotes all documents covered by term t, i.e., all documents in whichtappears, and|D(t)| represents the number of documents covered by t. A term t is frequent if |D(t)| ≥minsup, where minsup is a user-specified threshold, which represents the minimum number of documents in which a frequent term is required to occur. Each frequent term indicates a possible user interest. In order to organize all the frequent terms into a hierarchical structure, relationships between the frequent terms are defined below.Assuming two terms tA and tB., the two heuristic rules used in our approach are summarized as follows:

1.Similar terms: Two terms that cover the document sets withheavy overlaps might indicate the same interest. Here we usethe Jaccard function [27] to calculate the similarity between two terms: Sim(tA, tB) = | D(tA) D(tB) | / | D(tA)D(tB) |. If Sim(tA , tB) >δ, where δ is another user-specified threshold, we take tA and tB as similar terms representing the same interest.

2.Parent-Child terms: Specific terms often appear together withgeneral terms, but the reverse is not true. For example,"badminton" tends to occur together with "sports", but "sports" might occur with "basketball" or "soccer", not necessarily "badminton". Thus, tB is taken as a child term of tA if the condition probability P(tA | tB)>δ, where δ is the same threshold in Rule 1.

Rule 1 combines similar terms on the same interest and Rule 2 describes the parent-child relationship between terms. Since Sim(tA , tB) ≤ P(tA | tB), Rule 1 has to be enforced earlier than Rule 2 to prevent similar terms to be misclassified as parent-child relationship. For a term tA, any document covered by tA is viewed as a natural evidence of users' interests on tA. In addition, documents covered by term tB that either represents the same interest as tA or a child interest of tA can also be regarded as supporting documents of tA. Hence supporting documents on term tA, denoted as S(tA), are defined as the union of D(tA) and all D(tB),where either Sim(tA, tB) >δ or P(tA|tB) >δ is satisfied.Using the above rules, our algorithm automatically builds a hierarchical profile in a top-down fashion. The profile is represented by a tree structure, where each node is labeled a term t, and associated with a set of supporting documents S(t), exceptthat the root node is created without a label and attached with D, which represent all personal documents. Starting from the root, nodes are recursively split until no frequent terms exist on any leave nodes. Below is an example of the process.Before running the algorithm on the documents, preprocessing steps like stop words removal and stemming needs to be performed first. For simplification, each document is treated as a list of terms after preprocessing.

D1: sports, badminton
D2: ronaldo, soccer, sports
D3: sex, playboy, picture
D4: sports, soccer, English premier
D5: research, AI, algorithm
D6: research, adpative, personalized, search
D7: Fox, channel, sports, sex
D8: MSN, search
D9: research, AI, neuro network
D10: personalized, search, google, research

Figure 2. An example data source

Example 1.In Figure 1,10 documents are available as the datasource, from which the user profile will be built. The two parameters mentioned in Rule 1 and Rule 2 are set as minsup = 2, δ = 0.6.First, with a single scan of the documents, all frequent terms are sorted in a descending order of (document) frequency: <research: 4>, <sports:4>, <search:3>, <peronalized:2>, <soccer:2>, <AI:2>, <sex:2>. For each frequent term t, the initial supporting documents S(t) are set as D(t). All frequent terms are checked separately in a descending order of frequency. A node labeled term t is created if t satisfies neither Rule 1 nor Rule 2 with any other term t'. Supporting documents S(t) is attached with each node labeled t.

In this example, the term "research" was chosen first. This term applies to documents D5, D6, D9, and D10. A node labeled "Research" is created with S("Research")={D5, D6, D9, D10}. Similarly, a node labeled "sports" is generated with S("sports") ={D1, D2, D4, D7}. A merge operation arises when the term "search", which covers D6, D8 and D10, is examined. First, Sim("search" , "research") = 2/5 ≤δ is calculated. Then, P("research" | "search") = 2/3 >δ is checked. Since Rule 2 is satisfied, "search" is taken as a specific term under "research", and D("search") is merged

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

into S("research") . This is the same process for the terms "personalized" and "AI". Next, D("soccer") is merged into S("sports") since "soccer" is identified as a specific term under "sports". A new node is formed for term "sex", because both P("research"| "sex") =0 and P("sports"| "sex")=1/2 are less than δ. Three nodes "research", "sports" and "sex" are left after the merging operations. As we mentioned earlier, every document in S(t) is regarded as a supporting document of term t. And the support of term t, contributed by all documents in S(t), is anindication of the degree of the user's interest on t. For any document d in S(t), if d appears in n nodes (n≥1), which was interpreted as d supporting all n terms, the support from d in S(t) is counted only as 1/n.

This guarantees the sum of support contributed by each document equals to 1 in spite of the number of terms it supports. Thus the support of a term t, denoted as Sup(t), is calculated as the sum of the supports from all documents in S(t) . In this example, D7 appears in both S("sports") and S("sex"), so Sup("sports")=1+1+1+1/2=3.5, and Sup("sex") =1.5.A diagram of the user profile after the first splitting is shown in Figure 3, where the term t and its support Sup(t) are attached to each cluster, with the supporting documents S(t) listed below. Each node on the same level is sorted by Sup(t) in a descending order.

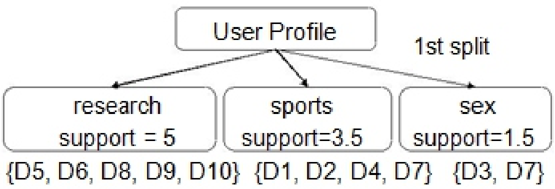

Figure 3.User profile after 1st split.

The node "research" is subsequently examined for further splitting. First S("research") is scanned, and the frequency for each term t is counted. Note that any term like "research" that appears in an ancestor node will not be counted again. Frequent terms and their frequency are listed as follows: <search:3>, <personalized:2>, <AI:2>. According to Rule 2, "search" and "personalized" is combined together and the node is labeled "personalized/search" since Sim("search", "personalized") = 2/3>δ. The child nodes after splitting are shown in Figure 4. The splitting can be recursively done until no term is frequent.

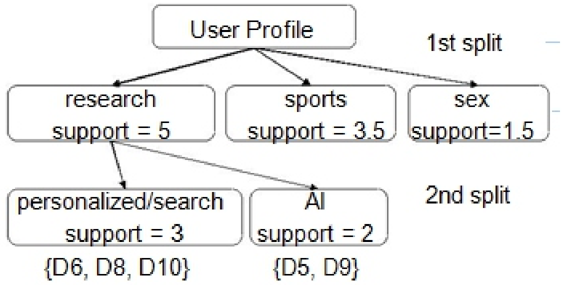

Figure 4.User profile after 2nd split

The formal algorithms are described in Figure 5. Split(n, S(t), minsup, δ) is called to split a node n. Rule 1 is enforced in line 3- 4,and Rule 2 is enforced in line 5-6. In line 9, nodes are sorted in a descending order of the support of term ti. The reason will be explained in section 4.2. A complete user profile is constructed by calling BuildUP(root, D, minsup,δ), where root represents the root node, and D is the set containing all personal documents. Split(n, S(t) minsup,δ) are recursively applied on each node until no frequent term exists on any leave node.

Algorithm: Split(n, S(t), minsup,δ)

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

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

Algorithm: BuildUP(n, D, minsup, δ)

Input: a node n, supporting documents D, thresholds minsup and δ Output: A user profile U

1.Split(n, D, minsup, δ)
2.for each child ci labeled ti of node n:
3.BuildUP(ci, S(ti), minsup, δ)

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

Figure 5. Algorithm for splitting a document set

4.2 Measuring Privacy:

According to Alan Westin [23], "privacy is the claim of individuals, groups, or institutions to determine for themselves when, how and to what extent information is communicated to others". Privacy per se is about protecting users' personal information. However, it is users' control that comprises the justification of privacy. With the complete user profile constructed above, an approach without any privacy risk is to grant users full control over the terms in the hierarchy so that they can choose to hide any terms manually as they desire. Unfortunately, studies have shown that the vast majority of users are always reluctant to provide any explicit input on their interests [24]. In order to offer users a more convenient way of controlling private information they would agree to have exposed, two parameters derived from information theory are proposed below.

In the following discussion, "interest" and "term" are indistinguishable in the context of the user profile. The support of an interest or a term t is Sup(t) , and S(t) represents all the supporting documents for term t. ΣSup(t)=|D| is for all terms t on the leave node, where |D| represents the total number of supports received from personal data.The user profile is established as an indicator of the users' possible individual interests. According to probability theories, the possibility of one interest (or a term) can be calculated as P(t)=Sup(t)/|D|. Within the context of information theory, the amount of information about a certain interest of the user is measured by its self-information [26]:

I(t) = log(1/P(t))= log(|D|/ Sup(t)), for any term t.

This measure has also been called surprisal by Myron Tribus[25] , as it represents the degree to which people are surprised to see a result. More specifically, the smaller Sup(t) is, the larger the self-information associated with the term t is, and more surprise occurs if the term t is exposed.Interestingly, this measure matches perfectly with our following observations on users' privacy concern: the interest with large self-information corresponds to two types of information to which users are usually sensitive to grant access to. The first case is that the interest itself is too specific. Users might not mind telling others about general interests, i.e. a user likes basketball, but is cautious about letting others know his weekly basketball schedule. The second case is that the interest is general but less popular among all interests. It might represent a private event, i.e. the category "sex" in Example 1. The idea is to protect private information that is either too specific or too sensitive in the user profile. Both kinds could be measured by the support of the interest, under the assumption that the more specific or sensitive the interest is, the larger self-information the interest will carry.This leads to the two parameters for specifying the requirement of privacy protection.

minDetail. The user profile above is organized from high-level tolow-level. Terms associated with each node become increasingly specific as the list progresses, and same level terms are sorted from left to right in descending order of their supports. A threshold of minDetail is defined to protect user's sensitive information on both vertical and horizontal dimensions. With a specified minDetail, any term t in the user profile with P(t)=Sup(t)/|D|<minDetail, will be protected from the server.Using Example 1, a fully extended user profile is shown in Figure 6, in which the dummy nodes labeled "others" are created to keep the user profile as a complete tree and to satisfy ΣSup(t)=|D| for all terms t on the leaf nodes. If minDetail = 0.3, details under the node "sports" are hidden, as well as "sex" that are on the same level with "sports", for P("sex") = Sup("sex")/|D|=1.5/10 < 0.3.

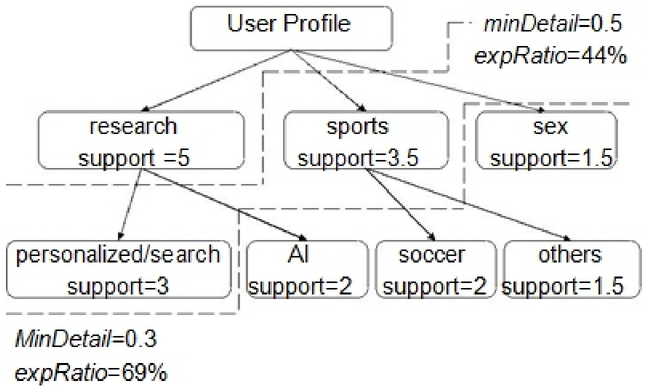

Figure 6. Fully extended user profile

The complete user profile is denoted as U, and U[exp] represents the exposed part of U, or the part above minDetail. Since the support for terms decreases monotonically traveling horizontally

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

and vertically, the U[exp] will be a connected subtree of the complete user profile stemming from the user profile root. With the threshold minDetail, the user will know exactly which part of the user profile is protected.expRatio. The thresholdminDetailfilters specific or sensitiveterms by their supports. Still, it is necessary to evaluate the "amount" of private information that is actually protected.For a given distribution of probabilities, the concept of entropy in information theory provides a measure of the information contained in that distribution [26]. We use entropy as a tool to calculate the amount of private information exposed by U[exp]. Consider a user's interest as a discrete random variable with probability mass function P(t), where t corresponds to any of a user's possible interests, and P(t)= Sup(t)/|D|. We denote by H(U[exp]) the entropy of U[exp], which can be calculated as:

$$H(U[exp]) = -\sum P(t) \times \log(P(t))t$$

wheret is any term on the leaves of U[exp]. Only the leaves are considered as the presence of terms on non-leaf nodes have already been counted by their children. Thus for any threshold minDetail, the exposed privacy can be calculated asexpRatio=H(U[exp])/H(U). Figure 7 shows U[exp] when minDetail is set as 0.3. Two leaf nodes labeled "others", which represent all the unexposed nodes, are added to maintain ΣSup(t)=|D| for all terms t on the leaf nodes. The actual terms are hidden since their support is less than 3. As the total support |D| is 10, it's possible to calculate H(U[exp])= - 0.3*log(0.3) - 0.2*log(0.2) - 0.35*log(0.35) - 0.15* log(0.15) = 0.580. It's also easy to calculate H(U)= 0.684 by considering all leaves in U (See Figure 6). Thus, expRatio= 0.580/0.684 = 69%.

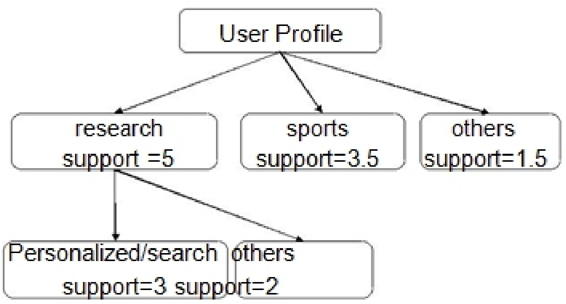

Figure 7. U[exp] when minDetail = 0.3 and expRatio = 69%

Two parameters, minDetail and expRatio, offer users the ability to determine the content and the amount of private information exposed.

As in the example, the lower the minDetail quotient, the more information that will be exposed, and expRatio will grow in relation to minDetail.The assumption behind two parameters is that more general and frequent terms, which carry smaller self-information, represent information users are more willing to share.

Nevertheless, we realize that it might not apply to some extreme cases. For example, a user may have a frequent and general interest in a sensitive topic (i.e. sexuality or politics) that he wants to keep private. Under this circumstance, a beneficial supplement to our solution is to allow users to hide certain branches of user profiles manually. However, more often than not, it is not necessary and a tedious work to most users. Our experiment results verified this.

4.3 Personalizing Search Results:

In order to incorporate the user profile with results returned by a search engine, U[exp] is transformed into a list of weighted terms where a search wrapper calculates a score for each of the returned search results. The final ranking of the search results is decided by the search engine and U[exp].The weight of each term in U[exp] is estimated by applying the concept of IDF(Inverse Document Frequency)Error! Reference source not found.. Given a termt, the weight of t, denoted bywt,is calculated as:

$$wt = \log(|D| / Sup(t)),$$

where |D| represents the total number of documents (or total support), and Sup(t) is the support of this term on the node in U[exp]. The partial user profile is expressed by a list <t, wt>, where t is a term in U[exp] and wt is the weight. Take U[exp] in Figure 7 as an example. The list is <research, 0.301>, <sports, 0.456>, <personalized/search, 0.523>.

The anonymous node labeled "others" is ignored.The workflow of personalizing web search results inside the search wrapper is illustrated in Figure 8. MSN Search is chosen as the search engine in our framework, and also in our experiments. A query is submitted to the search wrapper in four steps:

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

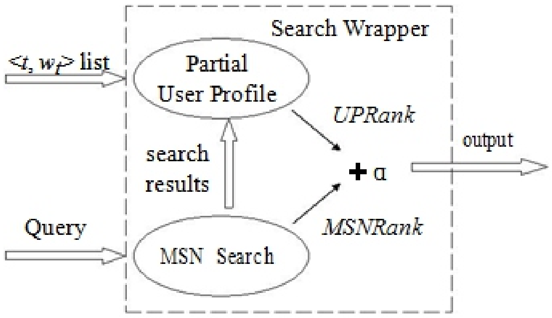

Figure 8. The workflow in the search wrapper

1.The user sends a query and the partial user profile to the search engine wrapper, where the partial user profile is represented by a set of <t, wt> pairs.
2.The wrapper calls the search engine to retrieve the search result from the web. Each result comprises of a set of links related to the query, where each link is given a rank from MSN search, called MSNRank. These links are passed to the partial user profile.
3.For each of the returned link l, a score called UPScore is calculated by the partial user profile as follows:

UPScore(l)=∑wt×tft

wheret is any term in the partial user profile, and tf is the frequency of the term t in the webpage of the link l. An UPRankis assigned to each link according to itsUPScore,and the link with the highest UPScore will be ranked first.
4.Re-ranking results by combining ranks from both MSN search and the partial user profile. The final rank, PPRank (Privacy-enhancing Personalized Rank), is calculated as:

PPRank=α*UPRank+ (1-α)*MSNRank,

where the parameter α[0, 1] indicates the weight assigned to the rank from the partial user profile. If α=0, the user profile is ignored, and the final rank is decided by the user profile instead of the search engine when α=1.

5. EXPERIMENTS:

In this section all experiments are conducted with the following objectives: to verify the effectiveness of the user profile to help improve search quality, and to explore the relationship between search quality and personal privacy.

5.1 Experiment Setup:

The approach is evaluated with 10 participants that run the client program on their own PC. Each participant built and viewed their own user profile, and issued their own queries by setting different parameters. In the user interface, three parameters could be adjusted: (1) personal data available for building a user profile– the choices given to the user were internet browsing history, emails, personal documents or any combinations thereof; (2) minDetail– the threshold offered to a user for determining whichpart of user profile is exposed. For any given minDetail, expRatiois updated to indicate the amount of information currently exposed; (3) α – the weight assigned to the user profile ranking. The queries evaluated were selected through two different methods, which were at the participants' discretion. In one approach, users were asked to select 25 queries from a list formulated to be general interests, i.e. aids, laptop, .net. In another approach, users were asked to choose 25 queries that mimic a search performed in daily life. The hypothesis was that this would allow for the capture of a user's search behavior in the real world. All participants were interns from different research groups in Microsoft, with high levels of computer literacy and familiarity with web search.

Web search results were first retrieved from MSN search engine. Due to the practical reason, we were not able to implement our search wrapper inside the current search engine, but on a proxy server instead. For each query the top 50 links returned from MSN search engine were re-ranked by the search wrapper and then returned to the user. We believe these include the most meaningful results, and retrieving more links will not have a major impact on the experiment results due to their low MSN search rankings.

Given a set of links returned for a query, the participant was asked to determine which in their opinion were relevant. The links were presented in a random order so as not to bias the participants. The queries with no result or with no links marked as relevant by users were ignored.To evaluate the search quality, we adopt a widely used measure, Average Precision [22], with a higher value indicating more relevant documents returned at an earlier time. Over a set of queries, search quality is represented by the mean of the average precisions, where Average Precision for a query is calculated as follows:

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

$$Average\ \ Precision = \sum_{i=1}^{n} \frac{i}{l_i.rank} \Big/ n$$

whereli the ith relevant links identified for a query, and n is the number of relevant links. Each relevant link li identified by participants will be associated with two ranks: PPRank which represents the final rank that combines both user profile and MSN search rankings, and MSNRank, which is the original MSN ranking. Average precision are calculated for both two different rankings. Intuitively, a higher average precision indicates a higher search quality.All programs were implemented in C#. The two parameters mentioned in section 4.1 are chosen empirically: minsup=5 (through which most of the meaningless words are filtered); δ =0.6. And all participants are advised to use the same parameters for the purpose of comparability.

5.2 Effectiveness of the User Profile:

First, it is a must to demonstrate the effectiveness of the user profile in helping customizing search results. The personal data options available in our program were browsing history, emails, and recent documents, where user can either choose one or any combination of these options. The average number of the types of personal data on all the participants' computers is listed in Table 1. The data entries without frequent terms were ignored.

Table 1.Average number of personal data.

Browsing histories	Emails	Recent documents
1060	605	29

In Figure 9, with all parameters fixed (minDetail=0, expRatio=100%,α=0.5), the comparison of the average precisionsfor the same group of queries, with different personal data options selected are shown. Compared to the original MSNRank, the average precision that incorporates the user profile is much higher, and the search quality improves. However, additional personal information does not always yield better results. The best search quality is achieved when data sources are set as browsing history and emails. The user profiles built from "all" personal data, including browsing history, emails and recent documents, have a similar performance to using only browsing history. Recent documents seem to have the negative effect on search quality because some of extremely lengthy documents introduce more noise than useful information.

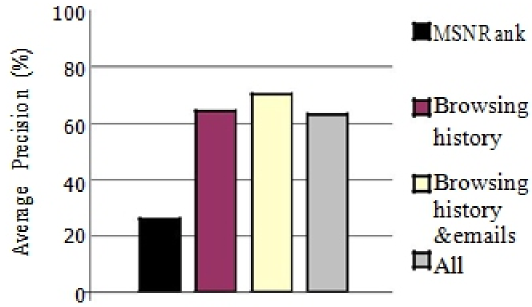

Figure 9.Effect of different personal data options.

Within the same group of queries, the impact of the user profile for PPRank is studied by varying only parameter α. The personal data options are set to browsing history & emails, minDetail = 0, and expRatio = 100%. Parameter α varies from 0 to 1, where α=1 indicates ranking search results by UPScore only, and α=0 shows the results from the original MSN search ranking.

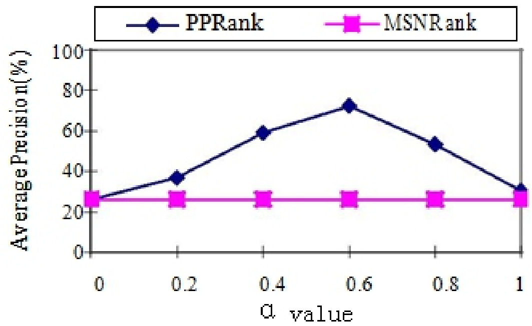

Figure 10. Impact of different αvalue

Figure 10 shows the average precisions of the PPRank, which depend on the user profile (α=1) and the original MSN ranking (α=0) respectively, are not acceptable. The best result occurs when α is around 0.6, and both ranks from MSN search and theuser profile are weighted almost equally. This indicates that the user's interest and the original ranking are both important to get better results.

5.3 Privacy vs Search Quality:

In this experiment, users are required to try different privacy thresholds to explore the relationship between privacy preservation and search quality. For each query, all parameters are fixed (personal data options are set to browsing history & emails, α = 0.6). expRatio will be updated in relation to a specified minDetail.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

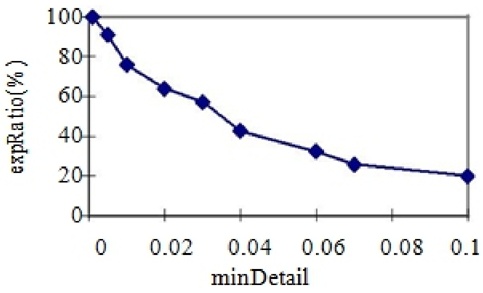

Figure 11.minDetailvsexpRatio

For any minDetail set by the user, the terms above the threshold will be exposed, and the remaining part of the user profile is protected from the search wrapper. The higher the minDetail is set, the less private information that is exposed leading to a smaller percentage of personal information exposed, or lower expRatio. The relation between minDetail and expRatio is illustrated in Figure 11. As minDetail increases, expRatio decreases almost linearly.

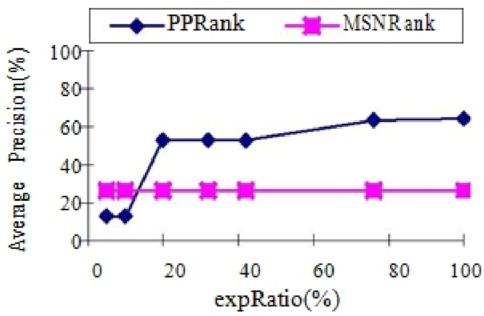

Figure 12.expRatiovs Search Quality

A group of search results is presented to show how search quality is affected by the amount of private information that is exposed. Figure 12 shows that the average precision of PPRank increased quickly when expRatio increased above 20%. However, as a user continues to expose more personal information the search quality only improves marginally.

There is almost no change when expRatioincreases from 80% to 100%.A case study from one of our participants demonstrates the reason that a small portion of privacy exposed could greatly increase search quality. When expRatio is set to about 20%, only 5 terms are exposed in the user profile. These include general interest terms like "research", "search", "sports" and websites frequently visited such as "Google" and "NYTimes".

Experiments showed that these general terms are especially helpful in identifying ambiguous queries like "conference" and "IT news". At the opposite extreme, over 100 terms are exposed when expRatio is set above 80%. Most of these terms indicate specific events that happened recently, such as "Winedown/Party" or websites that are occasionally visited (such as friends' blogs) which are too detailed to help refine the search. The experiment results above illustrate two points: first, general terms are much more useful than specific terms in helping to improve search quality. Second, too much private information exposed is not that useful. The experiments verify our hypothesis that exposing a small portion of our privacy could potentially return a relatively high search quality.

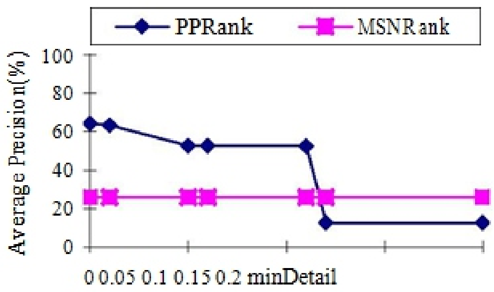

Figure 13.minDetailvs Search Quality

In Figure 13, the X-axis is changed to minDetail. This shows that hiding greater amounts of personal detail (minDetail from 0 to 0.1) does not decrease the search quality much. The most influential part of improving search quality is to use general terms with a minDetailabove 0.1.

5.4 Manual Privacy Option:

The aforementioned privacy parameters minDetail and expRatio, incorporating the hierarchical term-based user profile, offer users a convenient way to determine the extent to which personal information is exposed. This relies on the assumption that more general and frequent terms, which carry smaller self-information, represent information users are more willing to share. However, as we discussed in section 4.2, in some extreme cases a user may have a frequent and general interest in a sensitive topic that he wants to keep private. To solve this problem, the client program provides users the interface of hiding certain branches of user profiles manually. Consistently, any term labeled as private results in hiding all terms under this branch.

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

This facilitates a user who has to perform manual privacy option as he only needs to examine only a few high-level terms.The experiments show there are rare cases that users have the requirement of manually determining their private terms. Only 1 out of 10 participants has actually used this manual function. And the majority of participants prefer tuning minDetail into a larger value in order to meet their privacy requirements, rather than choosing to hide branches manually.

6. CONCLUSIONS AND FUTURE WORK:

Personalized search is a promising way to improve search quality. However, this approach requires users to grant the server full access to personal information on Internet, which violates users' privacy. In this paper, we investigated the feasibility of achieving a balance between users' privacy and search quality. First, an algorithm is provided to the user for collecting, summarizing, and organizing their personal information into a hierarchical user profile, where general terms are ranked to higher levels than specific terms. Through this profile, users control what portion of their private information is exposed to the server by adjusting the minDetailthreshold. An additional privacy measure,expRatio, isproposed to estimate the amount of privacy is exposed with the specified minDetail value. Experiments showed that he user profile is helpful in improving search quality when combined with the original MSN ranking. The experimental results verified our hypothesis that there is an opportunity for users to expose a small portion of their private information while getting a relatively high quality search. Offering general information has a greater impact on improving search quality.Yet, this paper is an exploratory work on the two aspects: First, we deal with unstructured data such as personal documents, for which it is still an open problem on how to define privacy. Secondly, we try to bridge the conflict needs of personalization and privacy protection by breaking the premise on privacy as an absolute standard. There are a few of promising directions for future work. In particular, we are considering ways of quantifying the utility that we gain from personalization, thus users can have clear incentive to comprise their privacy. Also, we suspect that an improved balance between privacy protection and search quality can be achieved if web search are personalized by considering only exposing those information related to a specific query.

7. REFERENCES:

[1]J. Pitkow, H. Schuetze, T. Cass, R. Cooley, D. Turnbull, A. Edmonds, E. Adar, and T. Breuel. Personalized search. Communications of the ACM, 45(9):50-55, 2002.

[2]Google personalized search: http://www.google.com/psearch

[3]Yahoo! My Web 2.0: http://myweb2.search.yahoo.com/

[4]W. Gasarch. A survey on private information retrieval. The bulletin of the European Association for TheoreticalComputer Science (EATCS), 82:72--107, 2004.

[5]Glen Jeh, and Jennifer Widom. Scaling personalized web search. In Proc. of the 12thInternational World Wide WebConference (WWW), Budapest, Hungary, May 2003.

[6]T.H. Haveliwala. Topic-sensitive PageRank. In Proc. of the11th International World Wide Web Conference (WWW),Honolulu, Hawaii, May 2002.

[7]K. Sugiyama, K. Hatano and M. Yoshikawa. Adaptive Web search based on user profile constructed without any effort from users, In Proc. of the 13thInternational World WideWeb Conference (WWW), New York, New York, May 2004.

[8]J.Teevan, S. T. Dumais, and Eric Horvitz. Personalizing search via automated analysis of interests and activities. In the Prof. of 28thAnnual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR), Salvador, Brazil. August, 2005

[9]Paolo Ferragina, and Antonio Gulli. A personalized search engine based on Web-Snippet hierarchical clustering. In Proc. of the 14th International World Wide Web Conference (WWW), Chiba, Japan, May 2005.

[10]P. A. Chirita, W. Nejdl, R. Paiu, and C. Kohlschutter. Using ODP metadata to personalize search. In the Prof. of 28thAnnual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR),Salvador, Brazil, August, 2005 .

ISSN No: 2348-4845
International Journal & Magazine of Engineering, Technology, Management and Research
A Peer Reviewed Open Access International Journal

[11]H.R. Kim, and Philip K. Chan. Learning implicit user interest hierarchy for context in personalization. In Proc. ofInternational Conference on Intelligent User Interface (IUI),Miami, Florida, January, 2003.

[12]M. Speretta, and S. Gauch, Personalizing search based on user search history. In Proc. of International Conference ofKnowledge Management(CIKM), Washington D.C., 2004

[13]P.Anick. Using terminological feed back for Web search refinement: a log-based study. In Proc. of the 13thInternational World Wide Web Conference (WWW), NewYork, New York, May 2004.

[14]K.R. McKeown, N. Elhadad, and V. Hatzivassiloglou. Leveraging a common representation for personalized search and summarization in a medical digital library. In Proc. ofInternational Conference on Digital Library, 2003

[15]A. Kritikopoulos, and M. Sideri. The compass Filter: Search engine result personalization using web communities. In Proc. of Intelligent Techniques in Web Personalization (ITWP), 2003.

[16]B. Fung, K. Wang and M. Ester. Hierarchical document clustering using frequent itemsets. In Proc. Of SIAMInternational Conference on Data Mining, San Francisco,May 2003.

[17]K. Wang, C. Xu, B. Ling, "Clustering transactions using large items", In Proc. of the 8th Conference on Informationand Knowledge Management (CIKM), Kansas City,November, 1999.

[18]J. Sun, H. Zeng, H. Liu, Y. Lu, and Z. Chen. CubeSVD: A Novel Approach to Personalized Web Search. In Proc. of the14th International World Wide Web Conference (WWW),Chiba, Japan, May 2005.

[19]R. Agrawal, and R. Skriant. Privacy preserving data mining. In Proc. of the ACM SIGMOD Conference on Managementof Data (SIGMOD), Dallas, Texas, May 2000.

[20]A. Evfimievski, J. Gehrke and R. Srikant. Limiting privacy breaches in privacy preserving data mining. In Proc. of theACM SIGMOD/PODS(PODS), San Diego, CA, 2003

[21]L. Sweeney. k-anonymity: a model for protecting privacy. International Journal on Uncertainty, Fuzziness and Knowledge-based Systems, 10 (5), 2002; 557-570

[22]R. Baeza-Yates, and B. Ribeiro-Neto, Modern InformationRetrieval. Addison Wesley Longman, MA, 1999.

[23]W. Alan. Privacy and Freedom. Atheneum Press, Boston, 19.