ISSN 2348-1196 (print)
International Journal of Computer Science and Information Technology Research ISSN 2348-120X (online)
Vol. 4, Issue 1, pp: (152-157), Month: January - March 2016, Available at: www.researchpublish.com

Secure and Proficient Web Search Using String Matching Algorithm

Deepika Sisode[1], Pooja Agrawal[2], Pratiksha Thorat[3], Chaitali Umap[4],
Prof. Arati Deshpande[5]

[1,2,3,4] UG Student, Dept. of Information Technology, JSCOE, Hadapsar, Pune, India
[5] Assistant Professor, Dept. of Information Technology, JSCOE, Hadapsar, Pune India

Abstract: **Web search engines (e.g. Google, Yahoo etc.) are widely used to find certain data among a huge amount of information in a nominal amount of time. However, these useful tools also pose a privacy threat to the end user's web search engines profile their end user's by storing and analyzing past searches submitted by them. In the introduced system, we can implement the String Similarity Matching Algorithm (SSM Algorithm) for improving the better search quality results. To address this privacy threat, present solutions introduce new mechanisms that introduce a high cost in terms of calculation and communication. Personalized search is a promising way to get better accuracy of web search, and has been attracting more attention recently. However, effective personalized search needs collecting and aggregating user information, which often increases serious concerns of privacy infringement for many users. Indeed, these concerns have become one of the main barriers for deploying personalized search applications, and how to do privacy-preserving personalization is a great challenge. In this we introduce and try to resist adversaries with broader background knowledge, such as richer relationship amongst topics. Richer relationship means we generalize the user profile results by using the background knowledge which is going to save in history. Through this we can hide the user search results. By using this mechanism, we can achieve the privacy.**

Keywords: **Privacy protection, Data Mining, Result retrival, Profile, generalization, Online Anonymity, IR evaluation, Automatic Identification.**

1. INTRODUCTION

The web search engine has long become the utmost important portal for ordinary people looking for useful information on the web. However, users might experience collapse when search engines return inconsistency results that do not meet their real intentions. Such insignificant largely due to the enormous variety of users' contexts and backgrounds, as well as the ambigous of texts. Personalized web search (PWS) [1] is a general category of search techniques aiming at providing better search results, which are estimate for individual user needs.

In this paper we present a novel protocol specially designed to protect the end users privacy in front of web search profiling [1].In this we propose and try to abide adversaries with broader background knowledge, such as richer relationship among topics. Richer relationship means we generalize the user profile results by using the background knowledge which is going to store in history [5]. Through this we can hide the user search results. In the Existing System, Greedy IL and Greedy DP algorithm, it takes massive computational and communication time. For generalize the retrieved data by using the background knowledge. Through this we can abide the adversaries. Privacy protection in publishing transaction data is an important problem. A key feature of transaction data is the extra sparsity, which renders any single technique ineffective in anonymizing such data. Among latter works, some incur high information loss, some result in data hard to interpret, and some undergo from performance drawbacks. This paper introduces to incorporate generalization and compression to reduce information loss. However, the combination is non-trivial. We introduce novel techniques to address the efficiency and scalability challenges. Data mining (the analysis step of the "Knowledge Discovery in Databases" process, or (KDD), an integrative subfield of computer science, is the computational process of determing patterns in large data sets involving ways at the interconnection of artificial intelligence, machine learning, statistics, and database systems. The overall goal of the data mining process is to selection information from a data set and transform it into an comprehensible structure for further use. Aside from the raw investigation step, it involves database and data management aspects, data pre-processing, model and inference consideration interesting metrics, complexity considerations, post-processing of discovered structures, visualization, and online refreshing. Generally, data mining (sometimes called data or knowledge discovery) is the process of searching data from different point of view and summarizing it into useful information that can be used to enlarge profits, cuts costs, or both. Data mining software is one of a number of analytical tools for analyzing data. It allows end user's to search data from many different dimensions or angles, categorize it, and summarize the relationships classified Technically, data mining is the process of discovering correlations or patterns among dozens of fields in large relational databases.

2. RELATED WORK

2.1 Supporting Privacy Protection in Personalized Web Search:

Personalized web search (PWS) is a generic category of search techniques aspiring at providing better search results, which are modified for individual user needs. As the cost, user information has to be bring togethered and analyzed to figure out the user intention behind the concerned query. The solutions to PWS can in ordinary be categorized into two types, namely click-log-based technique and profile-based ones [2]. The click-log based methods are straightforward they simply enforce bias to clicked pages in the user's query history. Although this strategy has been demonstrated to perform logically and considerably well it can only work on replicated queries from the same user, which is a strong limitation confining its applicability. In comparison profile-based methods upgrade the search experience with complicated user-interest models generated from user profiling approach. Profile-based approach can be potentially effective for almost all sorts of queries, but are addressed to be ambiguous under some circumstances.Although there are pros and cons for both types of PWS techniques,the profile-based PWS has determined more effectiveness in advancing the quality of web search.

The online phase handles queries as follows:

1. When a user issues a query qi 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 Gi satisfying the privacy requirements. The generalization process is guided by considering two conflicting metrics, namely 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 reranks them with the complete user profile.

2.2 Privacy Protection in Personalized Search:

In this paper, analytically survey the concern of privacy preservation in personalized search[4]. Here discriminate and represent four levels of privacy protection, and analyze abundant software architectures for personalized search. It shows that client-side personalization [1] has advantages over the existing server-side personalized search services in preserving privacy in this position; personalized web search cannot be done at the individual user level, but is possible at the group level. This may slow down the effectiveness of personalization because a group's information need to model an individual user's information. However, if the group is appropriately constructed so that people with similar interests are classed together, it has much richer user information to offset the sparse explanation of individual end user's information requirements. Thus the search performance may essentially be improved because of the availability of more information from the group profile. In this circumstance, personalized web search cannot be done at the distinct user level, but is possible at the group level. This may scale down the effectiveness of personalization because a group's information need description is used to model an individual end user's. However, if the group is properly created so that people with comparable interests are associated together, it may have much richer user information to offset the sparse explanation of specific user information needs. Thus the search performance may really be better because of the accessibility of more information from the group profile.

Server-side Personalization:

For server-side personalization [1] as shown in Figure 1(b), the personally identifiable information P(U) is stored on the search engine side. The search engine builds and updates the end user's profile either through the user's explicit[6] input (e.g., asking the user to mention personal interests) or by collecting the user's search history implicitly (e.g., query and click through history). Both approaches require the user to create an account to identify himself. But the latter approach requires no additional effort from the user and contains richer description of user information need.The advantage of this architecture is that the search engine can use all of its resources (e.g., document index, common search patterns) in its personalization algorithm. Also, the client software generally requires no changes. This architecture is accepted by some generic search engines.

Client-side Personalization:

For client-side personalization [1] as shown in Figure 1(c), the personally identifiable information is always stored on a end user's personal computer. As in the case of server-side personalization [2], the user profile can be created from end user's specification explicitly or search history implicitly. The client sends queries to the search engine and receives results, which is the same as in the general web search scenario. But given a user's query, a client-side personalized [2] search agent can do query expansion to generate a new query before sending the query to the search engine. The personalized search agent can also rerank the search results to match the inferred user preferences after receiving the search results from the search engine.With this architecture, not only the user's search behavior but also his contextual activities (e.g., web pages and personal information (e.g., emails, browser bookmarks) could be incorporated into the end user's profile, allowing for the construction of a much richer user model for personalization. The sensitive contextual information is generally not a big concern since it is strictly stored and used on the client side. Another benefit is that the over in computation and storage for personalization can be distributed among the clients.

2.3 Implicit User Modeling for Personalized Search:

In this paper, explicated how to assume a end user's interest from the end user's search context and practice the conditional implied user model for personalized search. A decision speculate basis and develop techniques for implicit user showing in information retrieval. They developed an intelligent client-side web search agent (UCAIR)[6] that can achieve eager implicit feedback, e.g., query development established on previous queries and immediate result reranking established on click through information. Research on web search show that search agent can progress search accuracy over the popular Google search engine. In this paper, described how to make and update a user model based on the instant search context and implicit feedback information and use the model to improve the accuracy of ad hoc retrieval. In order to extremely benefit the user of a retrieval system through implicit user modeling, offered to perform "eager implicit feedback". Those is, as soon as experimental any new piece of evidence from the user, and update the system's certainty about the user's information need and respond with improved retrieval outcomes based on the updated user model. A decision-theoretic basis for enhancing interactive information retrieval based on eager user model updating, in which the system replies to each achievement of the user by choosing a system exploit to enhance an efficacy function. In a traditional retrieval model, the retrieval problem is often to match a query with documents and rank documents giving to their relevance values. As a result, the whole retrieval progression is a simple independent cycle of "query" and "result display". In the planned new recovery model, the user's search circumstance shows a significant role and the conditional implicit user typical is exploited directly to benefit the user. The novel retrieval model is thus essentially diverse from the traditional pattern, and is inherently more general.

UCAIR: A PERSONALIZED SEARCH AGENT:

In this section, we present a client-side web search agent called UCAIR, in which we implement some of the methods discussed in the previous section for performing personalized search through implicit user modeling. UCAIR is a web browser plug-in 1 that acts as a proxy for web search engines. Presently, it is only implemented for Internet Explorer and Google, but it is a matter of engineering to make it run on other web browsers and interact with other search engines.

The UCAIR toolbar has 3 major components:

(1) The (implicit) user modeling module captures a user's search context and history information, including the submitted queries and any clicked search results and infers search session boundaries.

(2) The query modification module selectively improves the query formulation according to the present user model.

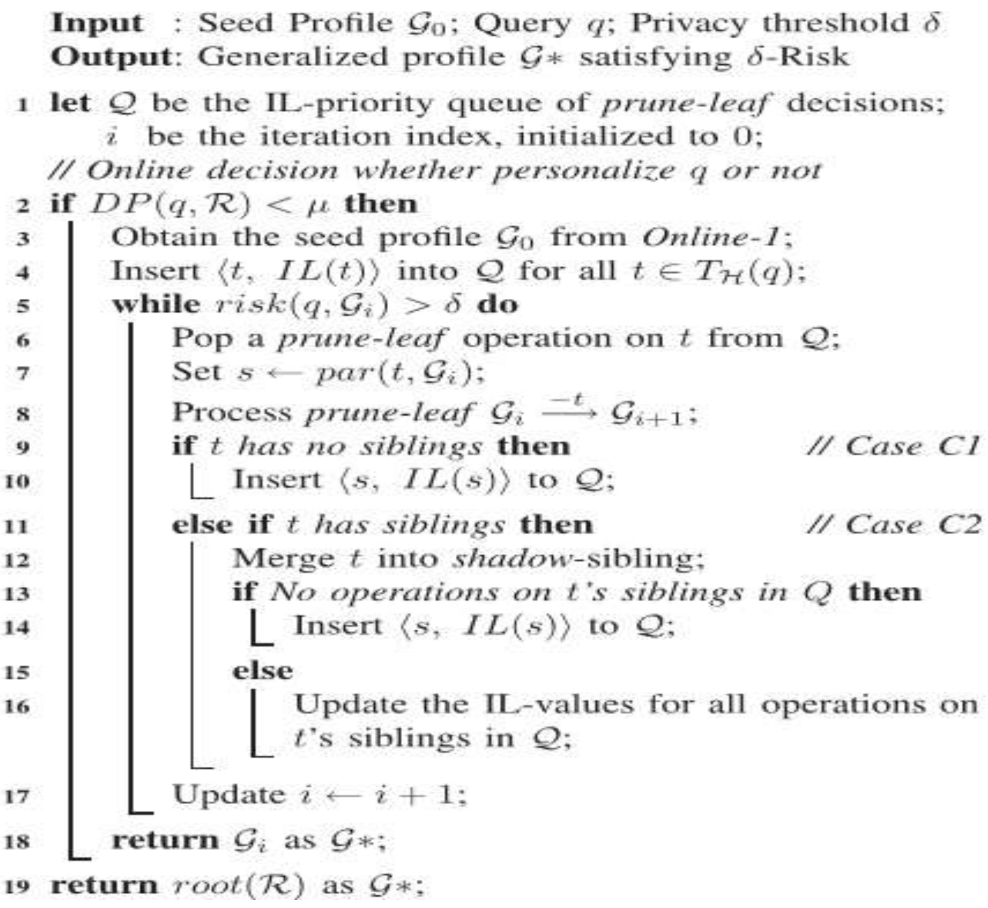
(3) The result re-ranking module immediately re-ranks any unseen search results whenever the user model is updated.

2.4 Online Anonymity for Personalized Web Services:

The notion of online anonymity to enable end user'sto issue personalized queries to an untrusted web service while with their anonymity preserved. The challenge for providing online anonymity is dealing with unknown and dynamic web end user'swho can get online and offline at any time. This problem, discusses its implications and differences from the problems in the literature, and introduces a solution. An algorithm that achieves online anonymity through the user pool. A significant challenge comes from the assumption of untrusted web service and user pool, and dealing with the dynamic sets of online users. Specifically, to provide online anonymity, the user pool must track the online end user'swho issued queries during a certain time interval and Anonymize their personal information d in an online fashion. This tracking also entails some interaction between the user pool and web users. A protocol for this interaction to guarantee that the additional information collected by the user pool cannot be used to compromise user anonymity. Although focus on anonymizing the personal information d that is separately provided for the personalization purpose, in the same spirit, approach can be extended to deal with personally identifying information that may be contained in the query d. In this sense, work is also applicable to general web services where there is a need to Anonymize the query, with or without personalization.

3. EXISTING SYSTEM ALGORITHM

Greedy IL Algorithm:



```
Input  : Seed Profile $\mathcal{G}_0$; Query $q$; Privacy threshold $\delta$
Output: Generalized profile $\mathcal{G}*$ satisfying $\delta$-Risk
let $\mathcal{Q}$ be the IL-priority queue of prune-leaf decisions;
    $i$ be the iteration index, initialized to 0;
// Online decision whether personalize q or not
if $DP(q, \mathcal{R}) < \mu$ then
    Obtain the seed profile $\mathcal{G}_0$ from Online-1;
    Insert $\langle t, IL(t) \rangle$ into $\mathcal{Q}$ for all $t \in T_{\mathcal{H}}(q)$;
    while $risk(q, \mathcal{G}_i) > \delta$ do
        Pop a prune-leaf operation on $t$ from $\mathcal{Q}$;
        Set $s \leftarrow par(t, \mathcal{G}_i)$;
        Process prune-leaf $\mathcal{G}_i \xrightarrow{-t} \mathcal{G}_{i+1}$;
        if $t$ has no siblings then                      // Case C1
            Insert $\langle s, IL(s) \rangle$ to $\mathcal{Q}$;
        else if $t$ has siblings then                    // Case C2
            Merge $t$ into shadow-sibling;
            if No operations on $t$'s siblings in $\mathcal{Q}$ then
                Insert $\langle s, IL(s) \rangle$ to $\mathcal{Q}$;
            else
                Update the IL-values for all operations on
                $t$'s siblings in $\mathcal{Q}$;
        Update $i \leftarrow i + 1$;
    return $\mathcal{G}_i$ as $\mathcal{G}*$;
return $root(\mathcal{R})$ as $\mathcal{G}*$;
```

4. PROPOSED SYSTEM ARCHITECTURE

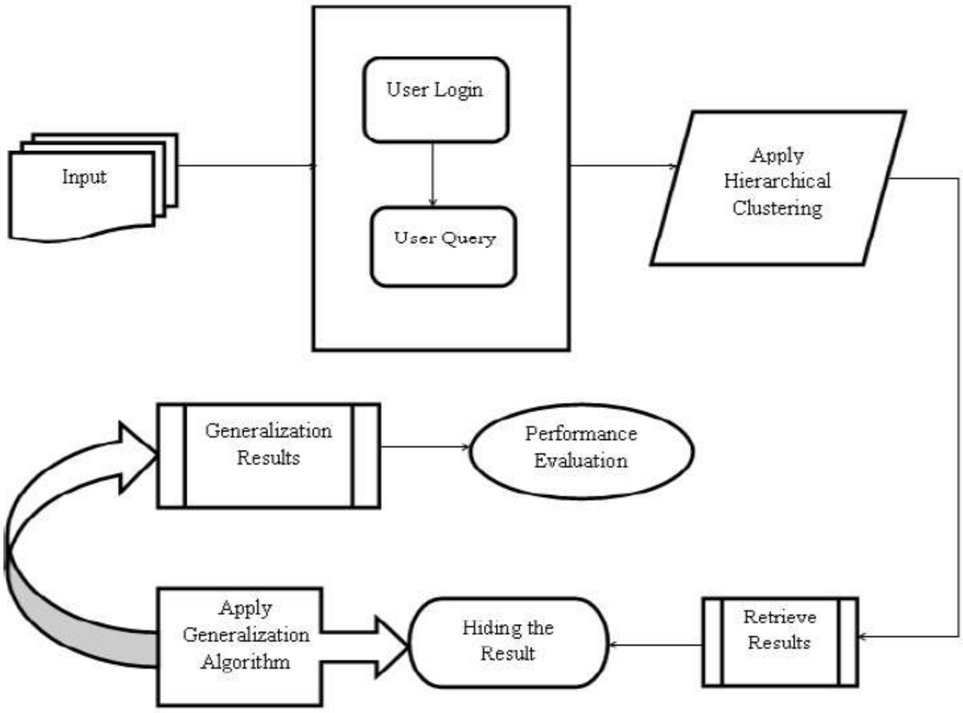


Fig. System Architecture

Modules:

- Dataset preprocessing
- User Login
- Query Searching and Search Results Retrieval
- Estimate Relevant Results
- Retrieve user profile in privacy manner

Module Descriptions:

1. Dataset preprocessing:

Most commonly a data set compare to the contents of a single database table, or a single statistical data matrix, where whole column of the table represents a particular variable, and each row corresponds to a given member of the data set in question. The data set lists values for all of the variables, such as height and weight of an object, for each member of the data set. Each value is called as a datum. The data set may comprise data for one or more members, corresponding to then umber of rows. In this module, choose input dataset. Choosen dataset has been loaded into the database. After loading the dataset into the database, we can view the dataset. By using the string matching algorithm we filter out unwanted values in the dataset and it has been preprocessed and store into the database.

2. User Login:

This is for user login page. In this module, end user'sare entered by using the unique id and password. In this module, end user's are entered after registering. After registering each end user's has unique id. After login, user posts some queries which is based on our dataset which is loaded into the database.

3. Query Searching and Search Results Retrieval:

In this module, user submits query. Based on the query, relevant results has been displayed and also based on the submitted query some history results are displayed. Based on the query and already posted queries, we can calculate the similarity values between them. In that three types of similarity values has been estimated. From that, we retrieve the result which is based on the high relevant results by using the nominal range of similar values.

4. Estimate Relevant Results:

In this module, user posts query and sub query also. Based on the query and sub query, estimate the results based n string matching. Based on the relevant results and total number of datas in the dataset, we can estimate the support values.

5. Retrieve user profile in privacy manner:

In this module, adversaries to mine the history results means, only query time has been displayed. In this, other information such as query, query results, username are not displayed by using the background knowledge. First we generalize the table, and then suppress the values based on the generalized table. Generalized values are stored in the history results. When the adversaries' views the history result means, they can only view the generalized results. Finally, we can evaluate the performance by using the parameter such as time, cost and communicational and computational cost.

5. CONCLUSION

For generalizing the retrieved data by using the background knowledge.Through this we can resist the adversaries. Privacy protection in publishing transaction data is an serious problem. A key feature of transaction data is the extreme sparsity, which provide any single technique ineffective in anonymizing such data. Among recent works, some incur high information loss, some result in data hard to clarify, and some suffer from performance drawbacks. This paper introduces to incorporate generalization and compression to reduce information loss. However, the integration is non-trivial. We introduce novel techniques to address the efficiency and scalability challenges.

Our introduced system gives better quality results and gives more efficiency. Privacy is too good when compared with the Existing system. In the Existing System, only generalization technique is used. Our String matching algorithm gives more accuracy when compared with the Greedy IL algorithm. Generalization and suppression technique achieves better privacy when compared with the existing system.

REFERENCES

[1] 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 YEAR 2014

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

[3] Y. Xu, K. Wang, G. Yang, and A.W.-C. Fu, "Online Anonymity for Personalized Web Services," Proc. 18th ACM Conf. Information and Knowledge Management (CIKM), pp. 1497-1500, 2009.

[4] X. Xiao and Y. Tao, "Personalized Privacy Preservation," Proc. ACM SIGMOD Int'l Conf. Management of Data (SIGMOD), 2006.

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

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

[7] 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.

[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] A. Viejo and J. Castell_a-Roca, "Using Social Networks to Distort Users' Profiles Generated by Web Search Engines," Computer Networks, vol. 54, no. 9, pp. 1343-1357, 2010.

[10] X.Xiao and Y. Tao, "Personalized Privacy Preservation," Proc.ACM SIGMOD Int'l Conf. Management of Data (SIGMOD), 2006.