IJARCCE

ISSN (Online) 2278-1021
ISSN (Print) 2319 5940

International Journal of Advanced Research in Computer and Communication Engineering
ISO 3297:2007 Certified
Vol. 5, Issue 7, July 2016

Securing User Protection in Personalized Web Search Using Slicing Algorithm

Chanda Agarwal[1], Prof. Dr. Suhasini Itkar[2]

M.E. Scholar, Computer Engineering, PES Modern College of Engineering, Pune, India [1]

Head of Computer Department, Computer Engineering, PES Modern College of Engineering, Pune, India [2]

Abstract: Web Search engines are designed to help their users to find information on the World Wide Web (WWW). However, 'generic' search engines are not capable to identify the specific needs of the individual user. These search engines have the same treatment to all users irrespective of their individual need for information. This problem can be solved using "Personalization". Personalization has the capability to give different search results for different users based on parameters like their interest, preference and knowledge. To meet this requirement, there is a need to capture user's personal information – therefore, certain users may not want to disclose it. Therefore, "privacy" is a major concern. To provide privacy to the user's information in Personalized Web Search application, a framework called "User Customizable Privacy-Preserving Search" is discussed. Two runtime generalizations of profiles called Greedy DP and Greedy IL are conversed and it is found that Greedy IL performs better than Greedy DP in terms of response time. Slicing technique is implemented at the proxy side for prevention from attacks that may disclose user's personal information.

Keywords: Personalized Web Search (PWS), Private Information Retrieval (PIR), Search Engine Web Search, Wrapper

I. INTRODUCTION

Web search engines are one of the most important portals for users trying to find useful information on the web. As the amount of information on the web is growing continuously at a fast pace, it has become increasingly difficult for the search engines to find the information relevant to a user. For example, for the query "office", some users may be searching for a vacant office space, while other users may be searching for popular Microsoft productivity software. Therefore, web search results should adjust to users with different information needs.

"Personalization" is the process of collecting, storing and analysing information about different visitors. One of the techniques used to achieve effective personalization is by having the visitors of a site fill in forms with information fields. The website then uses the database to match a user's need to the products or information provided at the site, with middleware facilitating the process by passing data between the database and the web site.

Consider an example of Amazon.com used for online book purchase/selling. One of the facilities provided by Amazon.com for its registered users is the suggestion of books, CDs depending on their previous purchase history or interest captured while browsing the Web. Customers tend to buy more when they know exactly what is available at the site and they do not have to hunt around for it. Cookies may be the most recognizable personalization tool. A "Cookie" is nothing but a piece of code that is sitting in a user's internet browser memory and informs the web site about a person. Using cookies, a Web site is able to identify its users by their name. Search engines like Google, Yahoo display richer information for some queries, including maps and weather (for location searches), reviews and prices (for product search queries) and profiles (for people searches). "Personalized Web Search" (PWS) is a method that helps users to search appropriate information based on their requirements. For finding this type of information, search engines like Google Search, Yahoo Search use user's information present in profile and earlier search histories. This process is called as "profile generalization". Profile generalization helps user to comb expected information by selecting appropriate words from user profile.

But the main disadvantage of this extraction is that without user's consent, secret data from the profile may be used for search which may violate the privacy of the user. A PWS is said to be efficient only if it provides privacy to the user's profile. The loss of privacy can be observed in shopping behavior where a customer provides frequent shopper card disclosing his detailed profile. Thus, people may compromise privacy for their economic benefit. To prevent from such unnecessary exposure of data, a method called "User Customizable Privacy Search" (UPS) that models the user preferences as hierarchical user profiles provides privacy for PWS applications. The UPS provides search results by adapting to the user's information needs and also provides privacy according to the user specified privacy requirements. This preserves user privacy as well as helps user to get exact information as per their requirement [1]. While doing the personalization, the problem is that the user might not want to disclose their

DOI 10.17148/IJARCCE.2016.5703

personal information. So, protecting user personal information is a major concern. Two generalization algorithm named Greedy DP and Greedy IL have been implemented and found that Greedy IL is better than Greedy DP in terms of time. An algorithm called slicing is implemented at the proxy side to give more protection to the user profiles.

II. EXISTING SYSTEM

Generic Search Engines are not able to identify the needs of individual users. There is no protection of user profiles provided at the proxy side. It has the following disadvantages:

1. User has to spend a lot of time in searching for the relevant document.
2. Generic Search engine returns the same result for different users for the same query.

III.PROPOSED SYSTEM

A technique called slicing is proposed at the proxy side to protect the user profiles. The key idea of slicing is to achieve better privacy. The slicing algorithm helps to increase the efficiency of the search and enhances user's privacy. Because of this addition the user profile becomes difficult to understand and the attacker has a minimal chance to identify the user. Since the attacker is unable to find or identify the user profile, the user is unable to access the hidden node-set of the particular user and thus the privacy of the user is now safe.

IV.LITERATURE SURVEY

J. Teevan et al [2] describes a method for personalized web search where he has done analysis of interest and activities of a particular user by using automatic user profile construction ranking algorithm. Results shows that text based personalization search algorithms perform better than relevance feedback method. M. Spertta and S. Gach,[3] examined the issue of privacy preservation in personalized search. Four levels of privacy protection are identified and analyze different software architectures for personalized search. This method shows that client-side personalization has advantages over the existing server-side personalized search services in preserving privacy. J. Castelli-Roca et al [4] presents a protocol called Useless User Profile (UUP) protocol, in order to protect the users' privacy in front of web search profiling. System provides a distorted user profile to the web search engine. This scheme also uses cryptographic building blocks such as Elgamal encryption, key generation, message encryption and decryption etc. for effective communication. The main idea of this scheme is that each user who wants to submit a query will not send her own query but a query of another user instead. At the same time, her query is submitted by another user. Using this approach, the web search engine cannot generate a real profile of a certain individual. The execution of queries may be delayed. The protocol assumes that, users follow the protocol correctly and no collision happens between entities, but in real it may be not the case. According to J. Pitkow et al.[5] there are mainly two techniques for search engines to search the information according to user's interest. First in contextualization. It searches according to information available on that topic, nature of search like web, pdf, images, files etc. and applications that uses that search. Second technique is individualization.

In this method user's goal, earlier search history is being considered and the user is presented with the search results. Z. Dou, R. Song, and J. R. Wen [6] describes two solutions to do Personalized Web Search namely click log base and profile base. Click log base method chooses search according to user's previous selected search. Profile based focuses on collecting the information from user's profile and produces the result according to that interest. It is found that profile based search is more suitable as compared to click based searches.

K. Sugiyama et al. [7] and X. Shen et al.[8] focuses to create a profile based on browsing history. Browsing history is acquired through following ways:

1. From user registration details which is filled up by user while profile formation.
2. From diverse queries that are submitted at the time of searching.
3. From results displayed by web search engines.

To improvise the privacy different levels like pseudo identity, group identity are suggested.

Leung et al. [9] proposes a new web search personalization approach that captures the user's interest and preferences by mining search results and their click through in the form of concepts. Ontology-based, multi-facet (OMF) user profiling strategy is used to capture the users content and location preferences for building a personalized search engine for mobile users. From the literature review, it has been found that OMF can provide more accurate personalized results comparing to the existing methods.

V. SYSTEM ARCHITECTURE

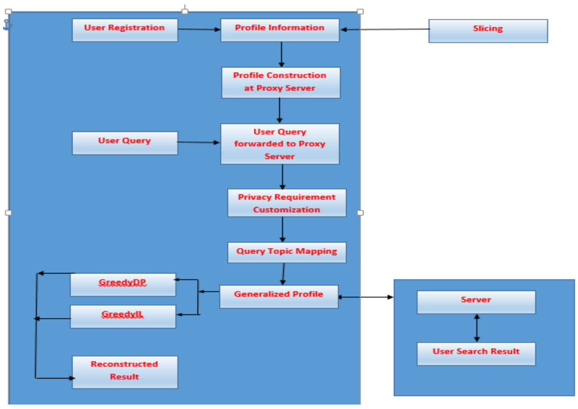


Figure 5.1: Architecture Diagram of the System

DOI 10.17148/IJARCCE.2016.5703

Existing System:
A. Algorithm used:
1. Greedy Discriminating Power (Greedy DP) Algorithm
2. Greedy Information Loss (Greedy IL) Algorithm
Each user has to undergo the following modules:
1. Profile Construction
2. Privacy Requirement Customization
3. Query-topic Mapping
4. Profile Generalization

User fired the query in the search engines and the results are returned from the server. Many times information loss occurs due to the algorithm inefficiency. Here, Greedy IL algorithm reduces the information loss during the information retrieval. The advantage of Greedy DP over Greedy IL is that the former requires more computation time due to lot of logarithmic operation being involved. It requires more computation time when the queries are ambiguous.

VI. ALGORITHMIC STRATEGY

A. Overall System Algorithm
1. CLIENT SIDE [FOR INPUT]
a) User Login
b) Authenticated User
c) User enter the query to be searched
d) Query along with the userid will be sent to the Proxy for processing
e) After the query is being processed by the proxy, results will be returned to the client
2. PROXY SIDE [FOR PROCESSING]
a) Load Profile Construction
b) Hide the user details from the attacker by implementing slicing algorithm
c) Find the sensitivity level of the sensitive node
d) Generate the Seed Profile for the search query
e) Calculate the generalized profile by using Greedy IL algorithm.

B. Greedy Algorithm
Input:
1. Seed Profile generated from Query-topic mapping module i.e. G_0
2. User Query i.e. q
3. Privacy threshold i.e. δ
Output:
Generalized Profile G satisfying the δ risk constraint
Steps:
1) Let prq be the IL priority queue containing user sensitive words
2) Initialize the loop counter i. ei to zero, μ to zero for identifying distinct queries, privacy threshold δ to 0.5
3) Calculate the discriminating power (DP) of the topic using the Greedy DP algorithm
a) Find the conditional probability of the query from a topic provided generalized profile is given using: Pr(t | q, G)
b) Calculate the discriminating power using the formula:

$$DP(t) = Pr(t \mid q, G) \log b \text{ where}$$

$$b = \frac{Pr(t \mid q,G)}{Pr(t)}$$

4) If DP(q,R) < μ then perform the following steps:
a) Obtain the seed profile generated from Query - topic mapping module
b) Insert this seed profile into the IL priority queue (prq)
5) Calculate the risk of query and seed profile using:

$$Risk(q,G) = \frac{Risk(root,G)\quad)}{\Sigma\, Sen(s)}$$

6) while risk(q,G_i) > σ do the following steps:
a) Calculate the sibling of each word from the seed profile
b) If sibling == 0 then
add this sibling to the priority queue
else
increment the counter
i = i + 1
endif
end while
7) Return the Generalized Profile and the result will be returned to the user

C. slicing Algorithm
Input:
User Information obtained from the database
Output:
User Profile in encrypted format
Steps:
1) Select the column to be encrypted from the user profile database
2) Select the encrypted symbol for each column with which you want to replace
3) Replace each column selected with the encrypted symbol
4) Display the user profile in encrypted format

VII. MATHEMATICAL MODEL

S = {I, M, O}
I = Search query entered by a particular user
Let M = {M_1, M_2,M_3,M_4} are different modules at Proxy side.
Where $M_1$is for the profile construction i.e..
P = {P_1, P_2,………….,P_n}are the profiles being created depending on the respective topic (here we are considering only the text file).
Let U = {U_1, U_2,……..,Un} are different registered users.
Let Q is the query which the particular user wants to search.
Then Q + U will be forwarded to the proxy server to determine which profile to be passed to the server.
M_2 is privacy requirement customization where
The sensitivity of each word is calculated for the particular user by using the formula:

a) For each sensitive node, cost(t) = sen(t)

DOI 10.17148/IJARCCE.2016.5703

b) For each non sensitive leaf node, cost(t)=0;
c) For each non sensitive internal node, cost(t) is recursively given by:

$$\text{Cost (t)} = \sum_{t'\varepsilon C(t,H)} \text{cost}(t')\ X\ \Pr(t' \mid t)$$

M_3 is Query topic mapping where for a particular query seed profile is searched in the profile (P) and generate seed profile G_0.
M_4 is Profile Generalization where
This G_0 will be passed to the profile Generalization and apply Greedy DP and Greedy IL algorithms and the results will be displayed at the client side.
• Find the Discriminating Power (DP) of Query and Repository = (Profile Granularity + Topic Similarity) / (Expected IC of Topics)
• Find the Profile Granularity of Query and Repository = $\sum\big(\Pr(q,G) * IC(t)\big) - H(q,G)$
• Find the Topic similarity of Query and Repository = IC(Ica(TG(q)))
• Find the Information Content IC(t) = log ^ - 1 * Pr(t)
• Find the Pr(t) = Pr(t | root(R))
O = {Personalized User Profile with Privacy maintained}

VIII. IMPLEMENTATION DETAILS

The system comprises of four modules:
1. Profile Construction
2. Privacy Requirement Customization
3. Query topic mapping
4. Profile Generalization

A. Profile Construction

• The first step of the offline processing is to build the original user profile in a topic hierarchy that reveals user interests.
• We assume that the user's preferences are denoted in a set of plain text documents.
• To build the profile, we take the following steps:
1. Detect the respective topic for every document Thus, the preference document set is transformed into a topic set.
2. Build the profile.

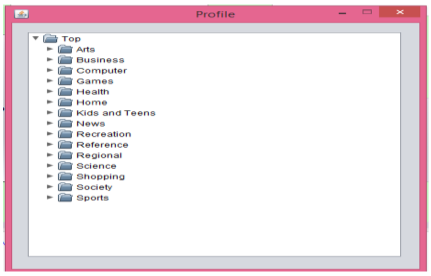


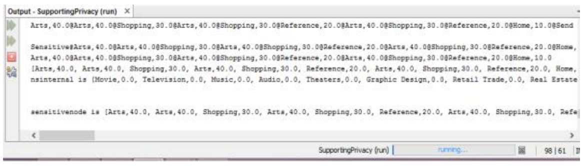
Figure 7.1: Profile Construction

B. Privacy Requirement Customization

• This process first requests the user to identify a sensitive-node set and the respective sensitivity value for each topic.
• Next, the cost layer of the profile is created by computing the cost value of each node using the following formula:

$$\text{Sensitivity} = \frac{tp}{tp+fn} * 100$$

Where tp is the total number of sensitive words and
fn is the number of sensitive words appearing in a
document

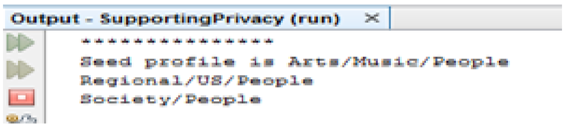

Output - SupportingPrivacy (run)

```
Arts,40.0@Arts,40.0@Shopping,30.0@Arts,40.0@Shopping,30.0@Reference,20.0@Arts,40.0@Shopping,30.0@Reference,20.0@Home,10.0@Send

Sensitive@Arts,40.0@Arts,40.0@Shopping,30.0@Arts,40.0@Shopping,30.0@Reference,20.0@Arts,40.0@Shopping,30.0@Reference,20.0@Home,
Arts,40.0@Arts,40.0@Shopping,30.0@Arts,40.0@Shopping,30.0@Reference,20.0@Arts,40.0@Shopping,30.0@Reference,20.0@Home,10.0
[Arts,40.0, Arts,40.0, Shopping,30.0, Arts,40.0, Shopping,30.0, Reference,20.0, Arts,40.0, Shopping,30.0, Reference,20.0, Home,
nsinternal is [Movie,0.0, Television,0.0, Music,0.0, Audio,0.0, Theaters,0.0, Graphic Design,0.0, Retail Trade,0.0, Real Estate


sensitivenode is [Arts,40.0, Arts,40.0, Shopping,30.0, Arts,40.0, Shopping,30.0, Reference,20.0, Arts,40.0, Shopping,30.0, Refe
```

Figure 7.2: Sensitivity value of Sensitive Node

C. Query Topic Mapping

Given a query, query-topic mapping does the following:
1. Find the topics in respective topic that are relevant to query.
2. Generate seed profile.

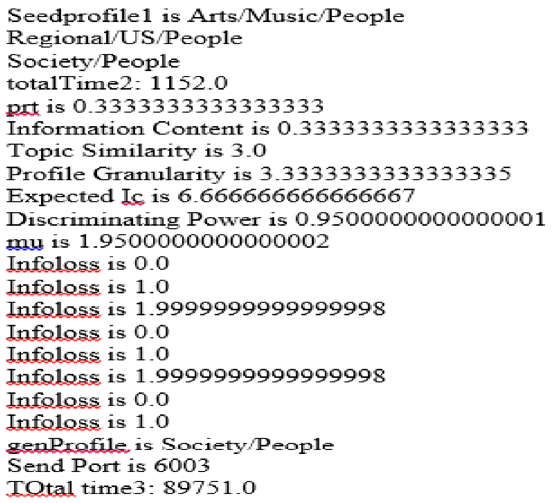

Output - SupportingPrivacy (run)

```
****************
Seed profile is Arts/Music/People
Regional/US/People
Society/People
```

Figure 7.3: Seed Profile Generated

D. Profile Generalization

• This procedure generalizes the seed profile in a cost-based iterative manner.
• This procedure computes the discriminating power for decision on whether personalization should be employed using greedy DP and greedy IL algorithm.



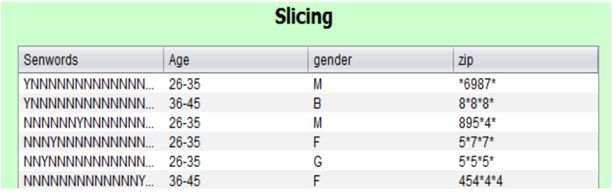
```
Seedprofile1 is Arts/Music/People
Regional/US/People
Society/People
totalTime2: 1152.0
prt is 0.3333333333333333
Information Content is 0.3333333333333333
Topic Similarity is 3.0
Profile Granularity is 3.3333333333333335
Expected Ic is 6.666666666666667
Discriminating Power is 0.9500000000000001
mu is 1.9500000000000002
Infoloss is 0.0
Infoloss is 1.0
Infoloss is 1.9999999999999998
Infoloss is 0.0
Infoloss is 1.0
Infoloss is 1.9999999999999998
Infoloss is 0.0
Infoloss is 1.0
genProfile is Society/People
Send Port is 6003
TOtal time3: 89751.0
```

Figure 7.4: Generalized Profile

The slicing algorithm helps to increase the efficiency of the search and enhances user's privacy. Because of this addition the user profile becomes difficult to understand and the attacker has a less chance to identify the user. Since the attacker is unable to find or identify the user

DOI 10.17148/IJARCCE.2016.5703

profile, the user is unable to access the hidden node-set of the particular user and thus the privacy of the user is now safe.



Slicing

Senwords	Age	gender	zip
YNNNNNNNNNNNNN...	26-35	M	*6987*
YNNNNNNNNNNNNN...	36-45	B	8*8*8*
NNNNNNYNNNNNNN...	26-35	M	895*4*
NNNYNNNNNNNNNN...	26-35	F	5*7*7*
NNYNNNNNNNNNNN...	26-35	G	5*5*5*
NNNNNNNNNNNNNY...	36-45	F	454*4*4

Figure 7.5: Slicing on User Profiles

IX. EXPERIMENTAL RESULTS

When we search for different queries, it has been found that Greedy DP takes more time as compared to Greedy IL which is shown in 8.1. The comparison is done based on their response time and information loss value.

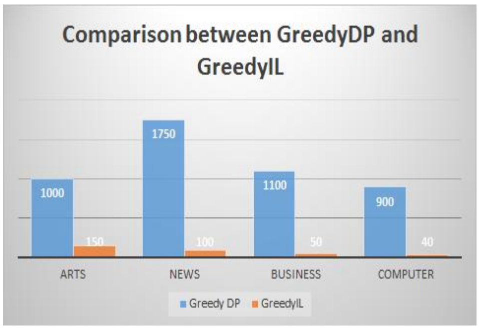


Figure 8.1: Comparison between Greedy dp & Greedy IL on Their Response Time

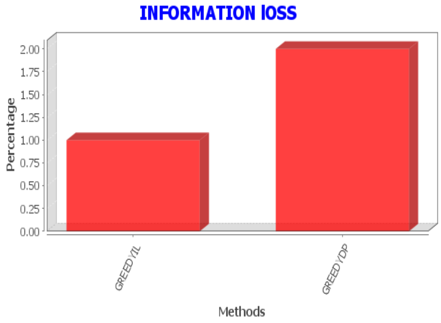


Figure 8.2: Comparison between Greedy DP and Greedy IL on their information loss value

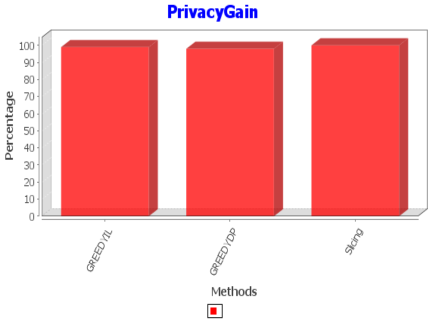


Figure 8.3: Privacy Gain

X. CONCLUSION AND FUTURE WORK

A client-side privacy protection framework called UPS for personalized web search IS PRESENTED. UPS could potentially be adopted by any PWS that captures user profiles in a hierarchical taxonomy. The framework allowed users to specify customized privacy requirements via the hierarchical profiles. In addition, UPS also performed online generalization on user profiles to protect the personal privacy without compromising the search quality. Two greedy algorithms, namely Greedy DP and Greedy IL, for the online generalization. Personalized search is a promising way to improve search quality. Because of the addition of slicing technique, the user profile becomes difficult to understand and the attacker has a less chance to identify the user. Since the attacker is unable to find or identify the user profile, the user is unable to access the hidden node-set of the particular user and thus the privacy of the user is now safe. In future, the framework may be applied for re-ranking the results retrieved by the search engines on the basis of user priorities. Collaborative filtering can also be applied for personalized web search.

REFERENCES

[1]. LidanShou, He Bai, Ke Chen and Gang Chen " Supporting privacy protection in personalized websearch" IEEE transaction on knowledge and data engineering vol:26 No:2 year 2014.

[2] J. Teevan, S. T. Dumais, and E. Horvitz., "Personalizing Search via Automated Analysis of Interests and Activities", Proc. 28th Ann. Intl ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR),pp. 449-456, 2005.

[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] J. Castelli-Roca and A. Vijeo and J. Herrera-Joancomarti, "Preserving user's privacy in web search engines, Computer Comm.vol.32,Vol.32,no.13/14, pp 1541-1551, 2009.

[5] 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.

[6] Z. Dou, R. Song, and J. R. Wen, "A Large Scale Evaluation and Analysis of Personalized Search," Proc. Int'l Conf. World Wide Web (WWW), pp. 581-590, 2007.

[7] 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.

[8] 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.

[9] Kenneth Wai-Ting Leung, Dik Lun Lee, Wang-Chien Lee. "Personalized Web Search with Location Preferences, " ICDE Conference 2010, pp. 701-712, 2010.

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

DOI 10.17148/IJARCCE.2016.5703