Improving the Efficiency of Personalized Web Search using Scoring Function

J. Anil [1], K. F. Bharathi[2]

[1]M.Tech (CSE), PG Scholar, Department of CSE, JNTUACEA

[2]Assistant Professor, Department of CSE, JNTUACEA

Abstract-- **Internet provides information regarding every discipline in the world. The discipline varies from traditional to technical. Irrespective of the subject internet provides information retrieving from the servers. In retrieval of required information search engines play vital role. Search engine follow different strategies in order to give the desired results as per the users requirement. But no search engine in the world results the required output as per the users wish. There are different types of search strategies like keyword & subjective search, phrase search, nesting and etc.**

In recent times privacy is being a major issue in the work of Lidan Shou and He Bai a web search framework UPS (User customizable Privacy-preserving Search) was used in order to generate output based on user interests. The framework utilises two major algorithms namely GreedyDP (Discriminating Power) and GreedyIL (Information Loss).

The search results can be enhanced by integrating the UPS framework with Scoring Function. The integrated framework gives us best results when compared to the work UPS framework.

I. INTRODUCTION

The web search engine has a huge amount of data related to any of the topics. If the people want any information first they approach the search engines. We have many search engines like Google, yahoo, MSN, etc. If a normal person wants the data about his/her interested topics the person enters the query based on that search engine gives the results.

Many times there is no possibility to get exact information within the time based on user interests. The web server gives the bulk amount of data, in that data he/she wants to check what information they want. It is a time tacking process to all netizens. For that Z.Dou, R.Song and J.R.Wen introduced the concept of personalized search strategies [2]. Here personalization means individual or particular. Personalized search provides information to the user from the web search engines (WWW) or from the web databases. The bulk amount of data divides into particular or individual concepts so here the users can get information easily.

After that not only divides the larger data into smaller once but also for improving search results introduce personalized search based on interests and activities [3] and also incorporate user search histories [4]. Many ways to creating user profile based on user information like profession, hobbies, interesting things, interesting topics he wants to search that type of information [5]. After that providing privacy to the user profiles and representation of user profiles can be implemented [8]. The user profiles based search predict means the search results from normal to specific search results [6]. For every unique user profile the search engine provides individual search results based on his interested queries. The personalized web search (PWS)[7] look critically or examine carefully for accuracy with the intent of verification for the data or results which obtained from the user profile. The search engine gives the data based on click-log based, URL domains and user clicks.

In today's era many people get information through web search, for all those users requires a user profile and providing security to those user profiles is a major issue. X. Shen and B.Tan proposed a privacy protection in personalized web search (PWS) [10].

Background definition

The supporting profile predicted personal web search does not purely works on runtime environment, it leads to indiscriminating the user requests for user profiles and also it is not protecting the user data. Hence data predicted individualize not only affect the quality of data but also exhibits normal data that gives normal search results based on database server.

For every user profile the server generates a new generalized user profile and checks the user query with user profile and gives the results. It is time taking process, re-computation of all candidate profiles and pruning the leaf needs to require more memory and computational cost.

II. RELATED WORK

In this section related working schema represented. It includes the model of user customizable privacy preserving search (UPS).

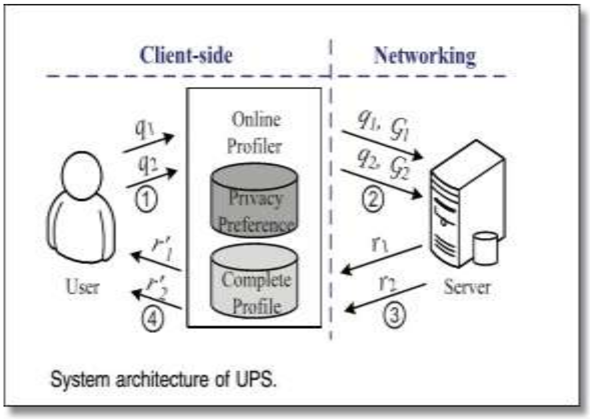


Fig 1: UPS architecture

A. UPS model of PWS

The above predicament/dilemma is represented in the UPS framework. The UPS consists of one server and number of users. Each user access the data from server. The UPS generates the online profiler. The online profiler is a search proxy and the proxy maintains the user profile with complete information.

1. The UPS framework works based on two stages that are offline stage and online stage for every candidate.
2. It gives the results based on general to specific.
3. There is no need of iterative user interaction for web search. Based on user profile the server gives the search results.

Whenever the candidate enters the query q_i on the user machine, the user proxy generates the generalized candidate profile at runtime in the lite of query terms. The candidate profile G_i fulfills the privacy demands or requisites. The generalization process is controlled by two conflict metrics those are personalization utility and privacy hazard. Subsequently the user query and generalized candidate profile combine together and send to the PWS server. The search results are compares with the candidate profile and give that result to the user proxy server. Finally the proxy gives results to the user.

$$G^* = \arg_G \max (util\ (q, G)),\ risk\ (q, G) < \delta$$

The profile based PWS focuses on improving the search quality. The basic idea behind the user profile is an individual information goal. The user profiles can be represented based on hierarchical structure. Weighted hierarchy graph [2][3][9]. The hierarchical structure user profile can constructed based on recent works, scalability and access efficiency.

The sample Taxonomy repository of the user profile

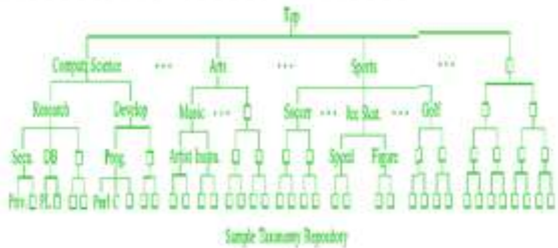


Fig 2: Taxonomy repository

nDCG (literally normalized discounted cumulative gain) is a common measure of information retrieval system.

B. Attack models

There are many attack models are there for our work majorly concentrate on providing security to the user profiles for privacy attacks namely eves dropping. Now Alice is a user he have his privacy information.

Alice communicates through PWS server. Eve is a attacker. The eavesdropper Eve successfully stops the communication between Alice and PWS server via some steps. This is called as man-in-middle attacks.

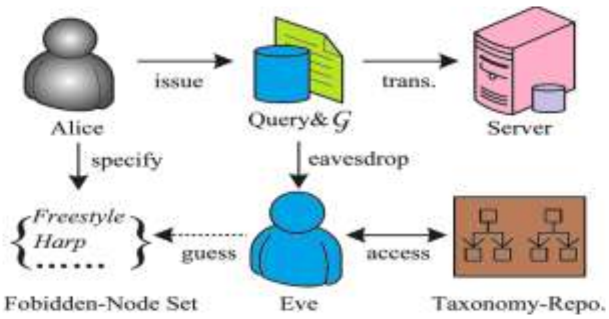


Fig 3: Attack models of PWS

Whenever user means Alice issue a query to the server the eve catches the query and generate runtime candidate profile based on that profile the eve touch the sensitive information nodes of a particular user.

- **Cognition Bounded**

  The knowledge from background learning of the opponent is limited taxonomy repository(*R*). The candidate profile *H* and privacy are anticipated on *R*.

- **Session Bounded**

  Previously caught information is available with eve for tracing the same victim in a long time. The above premise look strong and also reasonable because majority of privacy attacks are based on automatic programs to the large amount of PWS users.

III. PROPOSED WORK

This evolution includes the construction of new user profile based on user interested histories (UIH) and also score the terms which are entered by the user based on UIH by using scoring function.

User profile structure

The UPS follows the hierarchical structure representation and the user profile constructed based on user interested histories and availability of the public accessible taxonomy repository. *R* Is huge topic repository entropy and *t* is human recognizable topic.

$t \in R$

A. *GreedyDP, GreedyIL Algorithms*.

To find the more accurate results two optimal greedy algorithms are introduced. To find the accurate results and to provide more security to the user profiles uses above algorithms. In Greedy DP algorithm here introduce a concept called pruning mechanism. For pruning we use the operator →-t for pruning leafs of the hierarchical tree leaf nodes for getting more accurate results. H Is the hierarchical user profile and G_0 is the generalized user profile. Formally, denote by $G_{\text{i}} \to -tG_{\text{i+1}}$ the process of pruning leaf t from G_i to obtain G_{i+1}. Obviously the optimal profile G^* is the generative transitive closure of prune leaf. The Greedy algorithm improves the efficiency of user profile. After pruning leafs is there any information lost by the user. The prune leaf reduces the discriminating power of user profile. There are several findings on which heuristics depend on, to improve efficiency of GreedyIL algorithm. The critical finding is that the prune-leaf operation will cut down the cultivating supremacy of the profile. Otherwise, the DP displays monotonicity by prune-leaf operation. The DP theorem [1] states how the total computational cost will be reduced.

Scoring Function

In this evolution we first download the results from the Google/MSN search engines then use personalization strategy to give scores to the results.

1. Download top 50 search results from Google search engine. These results contain rankings of web pages.
2. The web page results which are downloaded from search engine compute personalization scoring that web page results.
3. Compare those scores and combine rankings then finalize the search results for the query is based on ranking fusion method.

The user interests can be very useful for personalizing the web search because it acts as a filtering aspect in searching rather than orthodox search.

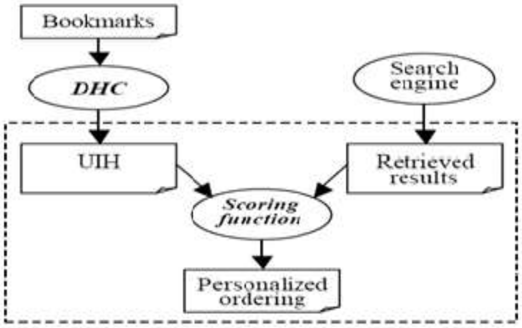


Fig 4: Personalizing search results through Scoring Function Architecture

A UIH is used to design scoring function for customized web search engine or commercial sites.

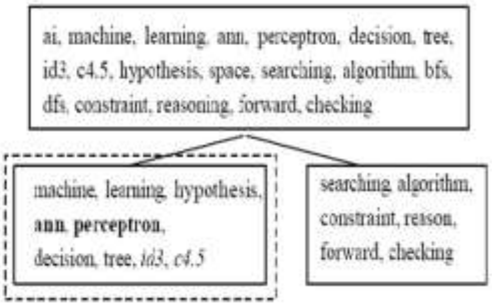
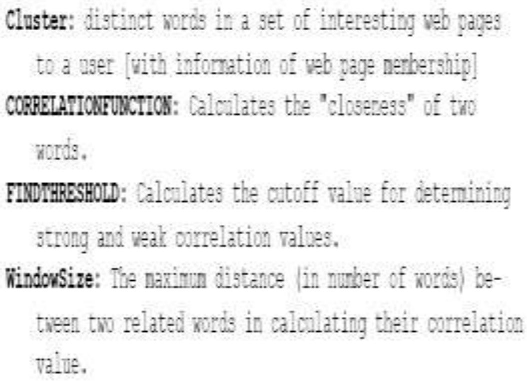

ai, machine, learning, ann, perceptron, decision, tree, id3, c4.5, hypothesis, space, searching, algorithm, bfs, dfs, constraint, reasoning, forward, checking
machine, learning, hypothesis, ann, perceptron, decision, tree, id3, c4.5
searching, algorithm, constraint, reason, forward, checking

Fig 5: UIH example

B. DHC algorithm

The following algorithm focuses on scoring a web page by considering user interests, which are measured in the form of scoring on web page.

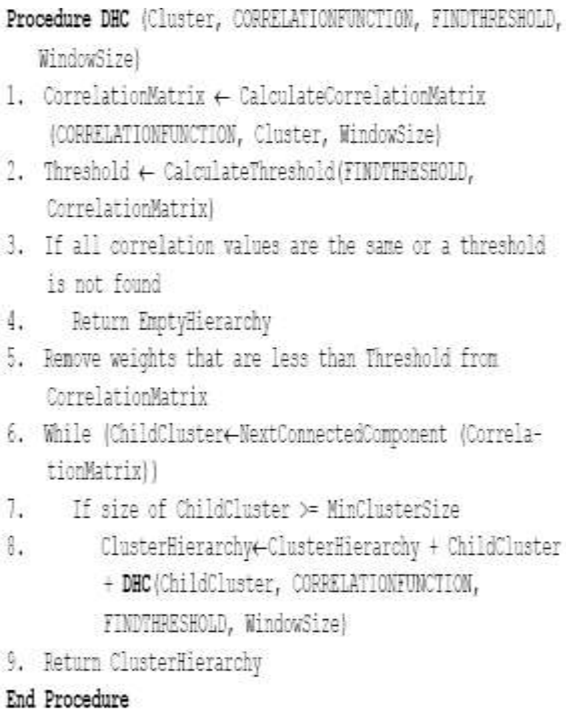


Cluster: distinct words in a set of interesting web pages to a user [with information of web page membership]

CORRELATIONFUNCTION: Calculates the "closeness" of two words.

FINDTHRESHOLD: Calculates the cutoff value for determining strong and weak correlation values.

WindowSize: The maximum distance (in number of words) between two related words in calculating their correlation value.



```
Procedure DHC (Cluster, CORRELATIONFUNCTION, FINDTHRESHOLD,
  WindowSize)
CorrelationMatrix ← CalculateCorrelationMatrix
  (CORRELATIONFUNCTION, Cluster, WindowSize)
Threshold ← CalculateThreshold(FINDTHRESHOLD,
  CorrelationMatrix)
If all correlation values are the same or a threshold
  is not found
    Return EmptyHierarchy
Remove weights that are less than Threshold from
  CorrelationMatrix
While (ChildCluster←NextConnectedComponent (Correla-
  tionMatrix))
    If size of ChildCluster >= MinClusterSize
        ClusterHierarchy←ClusterHierarchy + ChildCluster
        + DHC(ChildCluster, CORRELATIONFUNCTION,
        FINDTHRESHOLD, WindowSize)
Return ClusterHierarchy
End Procedure
```

C. Scoring Function

The page scoring depends on the factors like the count of interesting terms and how interesting the terms are in the web page. $S_{p_j} = \sum_{i=1}^{m} S_{t_i}$

m is the total number of matching terms, S_{ti} is the score for each distinct term and S_{pj} is the score for each distinct page.

IV. EXPERIMENTAL RESULTS

In this experiment we evaluate the efficiency of the search quality in inferior search engines using UPS framework. The search results are score based on UIH over 100 users, reduce the computational cost and improve the quality of search results.

Scoring based on	Average scoring	Improvement in %
Google(original)	4.8	---
Conceptual(queries)	3.4	36%
Scoring function	2.8	41%

Table 1: Comparison of scoring pages for validation queries

V. CONCLUSION

During extraction of various terms from a webpage, our work aims to derive to provide rank/score to the results from the search engine with UIH, which is a learned user profile framework. In such a way, this papers work determines the user interests for concerned web pages are formed implicitly and there is no need for asking the user directly. However, there is further scope for improving the efficiency.

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, February 2014.
2. Z. Dou, R. Song, and J.R. Wen, "A Large-Scale Evaluation and Analysis of PersonalizedSearch Strategies," Proc. Int'l Conf. World Wide Web (WWW), pp. 581-590, 2007.
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. X. Shen, B. Tan and C. Zhai, "Implicit user modeling for personalized search," Proc. 28th Ann. Int'l ACM SIGIR conf. Research and Development Information Retrieval (SIGIR), 2005.
5. F. Qui and J. Cho, " Automatic Identification of User Interest for personalized search," Proc. 15th Int'l Conf. World Wide Web (WWW), pp. 727-736, 2006.
6. J. Pitkow, H. schutze, T. cass, R.Cooley, D.Turnbull, A. Edmonds, E. Adar, and T.Breuel, "personalized Search," Comm. ACM, vol. 45, no. 9, pp.50-55, 2002.
7. A.Krause and E. Horvitz, "A Utility-Theoretic Approach to Privacy in Online Services," J. Artificial Intelligence Research, vol. 39, pp. 633-622, 2010.
8. A.Pretscner and S. Gauch, "Ontology-Based Personalized Search and Browsing," Proc. IEEE 11th Int'l Conf. Artificial Intelligence (ICTAI '99), 1999.
9. 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.

J. ANIL has obtained B. Tech degree in Computer Science and Engineering from JNT University A.P, India. Currently pursuing M.Tech in Software Engineering from JNTUACEA Ananthapuramu.
E-mail: anilkumar.jamalla@gmail.com

Dr. K. F. Bharati is currently working as an Assistant Professor in the Department of Computer Science and Engineering in JNTUA College of Engineering anathapuramu, A.P., India. She has received her Ph.D from JNTU Anatapur. She obtained her M.Tech from Visveswaraiah Technological University, Belgaum. She did her B.Tech from university of Gulbarga.