ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

IJESRT

INTERNATIONAL JOURNAL OF ENGINEERING SCIENCES & RESEARCH TECHNOLOGY

Enhanced Privacy Protection in Personalized Web Search for Sequential Background

N. Kalaivani* and Dr.P.Krishnakumari
* M Phil Research Scholar, RVS College of Arts & Science, Coimbatore , India
Director MCA Department, RVS College of Arts & Science, Coimbatore , India

Abstracts

Personalized Web Search has established to improve the quality of various search services on the Internet. Due to the tremendous data opportunities in the internet the privacy protection is very important to preserve user search behaviours and their profiles. In the existing system two generalized algorithms named as GreedyDP and GreedyIL were applied to protect private data's in Personalized Search Engine. The existing systems failed to resist sequential and background knowledge adversaries who has the broader background knowledge such as richer relationship among topics. The proposed introduces vector quantization approach piecewise on the datasets which segmentize each row of datasets and quantization approach is performed on each segment, using the proposed approach which later are again united to form a transformed data set. The proposed work is implemented using MATLAB and is analyzed using certain parameters such as Precision, Recall, Frequency Measure, Distortion and Computational Delay.

Keywords: Privacy protection, personalized web search, profile, vector quantization.

Introduction

The web search engine is the most important portal for ordinary people looking for useful information on the web. However, users generally experience failure and get improper results when search engines return irrelevant results that do not meet their real intentions. A typical search engine provides similar set of results without considering of who submitted the query. Therefore, the requirement arises to have personalized web search system which gives outputs appropriate to the user as highly ranked pages. Personalized web search (PWS) is a general category of search techniques which aims to provide better search results, according to individual user needs. So, for this user information has to be collected and analyzed so that the perfect search results required for the user behind the issued query is to be given to the user. The solution to this is Personalized Web Search (PWS).

It can generally be categorized into two types, first is click-log-based methods and second is profile-based ones. The click log based methods are simple and straightforward: This method performs the search based upon clicked pages in the user's query history. Although this method has been demonstrated to perform consistently and considerably well , it can only work on repeated queries from the same user, which is a strong limitation and restricted for certain applications. In contrast, profile-based methods improve the search experience with complicated user-interest models generated from user profiling techniques. Profile-based methods can be proved more effective for almost all sorts of queries, but are reported to be improper under some situations. Although there are reasons and considerations for both types of PWS techniques, the profile-based PWS has proved its more effectiveness in improving the quality of web search recently, with increasing usage of one's personal and behavioral information to profile its users, which is usually gathered implicitly with the help of query history, browsing history, click-through data, bookmarks, user documents and so on. Unfortunately, such type of collected personal data can easily reveal a entire scope of user's private life.

Related work

Search personalization is based on the fact that individual users tend to have different preferences and that knowing the user's preference can be used to improve the relevance of the results the search engine returns. There have been many attempts to personalize web search. These attempts usually differ in

- How to infer the user preference, whether explicitly by requiring the user to indicate information about herself or implicitly from the user's interactions,
- What kind of information is used to infer the user's preference,
- Where this information is collected or stored, whether on the client side or the server side, and
- How this user preference is used to improve the results' retrieval accuracy.

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

Lidan Shou, et.al, 2014, [1] presented a client-side privacy protection framework called UPS for personalized web search. 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.

Zhicheng, et.al, 2007, [6] proposed personalized search has been used for many years and many personalization strategies have been investigated, it is still unclear whether personalization is consistently effective on different queries for different users, and under different search contexts. The paper studies the problem and provides some preliminary conclusions. The paper present a large-scale evaluation framework for personalized search based on query logs, and then evaluate five personalized search strategies (including two click-based and three profile-based ones) using 12-day MSN query logs. By analyzing the results, it reveal that personalized search has significant improvement over common web search on some queries but it has little effect on other queries (e.g., queries with small click entropy).

Susan T. Dumais, et.al, 2005, [14] proposed search algorithms that consider a user's prior interactions with a wide variety of content to personalize that user's current Web search. Rather than relying on the unrealistic assumption that people will precisely specify their intent when searching, it pursues techniques that leverage implicit information about the user's interests. This information is used to re-rank web search results within a relevance feedback framework. It explore rich models of user interests built from both search-related information such as previously issued queries and previously visited web pages and other information about the user such as documents and email the user has read and created. The research suggests that rich representations of the user and the corpus are important for personalization but that it is possible to approximate these representations and provide efficient client-side algorithms for personalizing search. Such personalization algorithms can significantly improve on current web search.

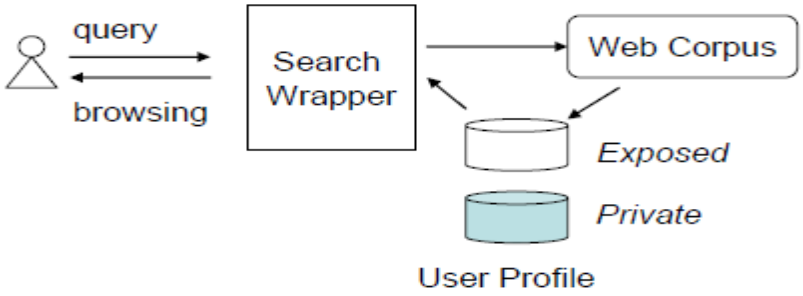
Jordi Castella-Roca, et.al, 2010, [15] proposed the Internet is one of the most important sources of knowledge in the present time. It offers a huge volume of information which grows dramatically every day. Web search engines (e.g. Google, Yahoo...) are widely used to find specific data among that information. However, these useful tools also represent a privacy threat for the users: the web search engines profile them by storing and analyzing all the searches that they have previously submitted. To address this privacy threat, current solutions propose new mechanisms that introduce a high cost in terms of computation and communication. The paper proposes a new scheme designed to protect the privacy of the users from a web search engine that tries to profile them. The system uses social networks to provide a distorted user profile to the web search engine. The proposed protocol submits standard queries to the web search engine; thus it does not require any change in the server side.

Existing design

The existing profile-based Personalized Web Search does not support runtime profiling. A user profile is typically generalized for only once offline, and used to personalize all queries from a same user indiscriminatingly. Such "one profile fits all" strategy certainly has drawbacks given the variety of queries. One evidence reported in is that profile-based personalization may not even help to improve the search quality for some ad hoc queries, though exposing user profile to a server has put the user's privacy at risk.

The existing methods do not take into account the customization of privacy requirements. This probably makes some user privacy to be overprotected while others insufficiently protected. For example, in, all the sensitive topics are detected using an absolute metric called surprisal based on the information theory, assuming that the interests with less user document support are more sensitive. However, this assumption can be doubted with a simple counterexample: If a user has a large number of documents about "status," the surprisal of this topic may lead to a conclusion that "status" is very general and not sensitive, despite the truth which is opposite. Unfortunately, little prior work can effectively address individual privacy needs during the generalization.

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114



Existing Design Structure

The above figure 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 and specific interests are put on a lower level. Only some 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.

Algorithm of Existing Design

Assuming two terms tA and tB., the two heuristic rules used in existing design are

Rule 1: Two terms that cover the document sets with heavy overlaps might indicate the same interest. The Jaccard function is used 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, take tA and tB as similar terms representing the same interest.

Rule 2: Specific terms often appear together with general 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.

The existing design algorithm consists of two stages called Split and BuildUp. The following steps describes the Split process of User profile

Step 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.

Step 2: The List of user profile entries is ordered using ascending or descending based on the value of the user.

Step 3: 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 search, called SearchRank. These links are passed to the partial user profile.

Step 4: For each of the returned link l, a score called UPScore is calculated by the partial user profile as follows: (Σ×=tttfwlUPScore) where t 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 UPRank is assigned to each link according to its UPScore, and the link with the highest UPScore will be ranked first.

Step 5: The similarity of user terms can be identified and that covers the document sets with overlap of the user profile.

Step 6: The specific terms often appear together with general terms of the user profile and it can be split based on the rank of the user list.

Step 7: 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.

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 interest and term. The following steps describes the BuildUp process of User profile

Step 1: "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.

Step 2: ΣSup(t)=|D| is for all terms t on the leave node, where |D| represents the total number of supports received from personal data.

Step 3: According to probability theories, the possibility of one interest (or a term) can be calculated as P(t)=Sup(t)/|D|. The amount of information about a

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

certain interest of the user is measured by its self-information I(t) = log(1/P(t))= log(|D|/ Sup(t)), for any term t.

Step 4: 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.

D represents the collection of all personal documents and each document is treated as a list of terms. D(*t*) denotes all documents *covered* by term *t*, i.e., all documents in which *t* appears, 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.

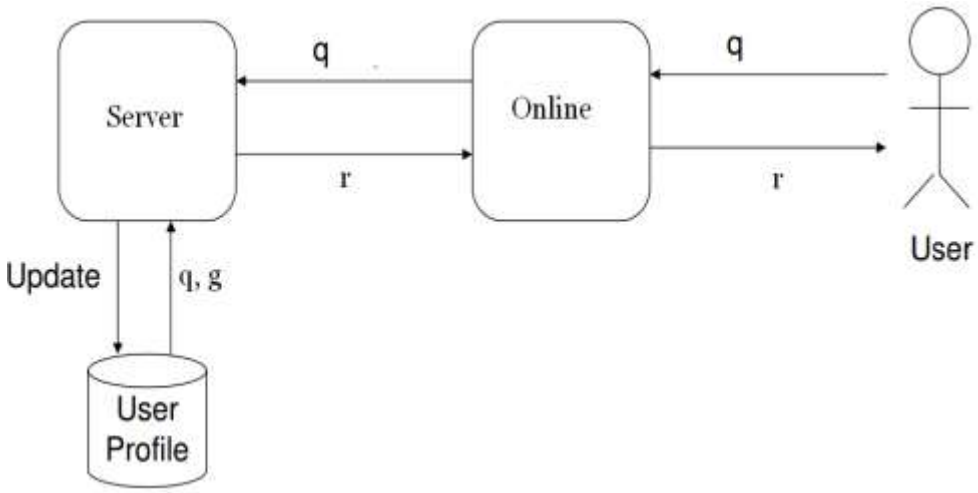
Using the above steps, existing algorithm automatically builds a hierarchical profile in a top-down fashion. The profile is represented by a tree structure, where each node is labeled a term *t*, and associated with a set of supporting documents S(*t*), except that the root node is created without a 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.

Drawbacks of existing design

- The existing profile-based PWS do not support runtime profiling
- The existing methods do not take into account the customization of privacy requirements
- Many personalization techniques require iterative user interactions when creating personalized search results.
- The existing system suffers from the customized privacy policy maintenance.
- Privacy protection domain requires iterative user interactions for personalization. This produced ineffective results.
- Failed to protect data from sequential and background attackers.

Proposed design

The proposed design contains a privacy-preserving personalized web search framework UPS, which can generalize profiles for each query according to user-specified privacy requirements. Relying on the definition of two conflicting metrics, namely personalization utility and privacy risk, for hierarchical user profile, we formulate the problem of privacy-preserving personalized search as Risk Profile Generalization, with its NP-hardness proved. It has two simple but effective generalization algorithms, GreedyDP and GreedyIL, to support runtime profiling. While the former tries to maximize the discriminating power (DP), the latter attempts to minimize the information loss (IL). By exploiting a number of heuristics, GreedyIL outperforms GreedyDP significantly. We provide an inexpensive mechanism for the client to decide whether to personalize a query in UPS. This decision can be made before each runtime profiling to enhance the stability of the search results while avoid the unnecessary exposure of the profile.



Structure of Proposed Design

Profile based personalization

An approach to personalize digital multimedia content based on user profile information. For this, two main mechanisms were developed: a profile generator that

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

automatically creates user profiles representing the user preferences, and a content-based recommendation algorithm that estimates the user's interest in unknown content by matching her profile to metadata descriptions of the content. Both features are integrated into a personalization system.

Privacy protection in PWS system

We propose a PWS framework called UPS that can generalize profiles in for each query according to user-specified privacy requirements. Two predictive metrics are proposed to evaluate the privacy breach risk and the query utility for hierarchical user profile. We develop two simple but effective generalization algorithms for user profiles allowing for query-level customization using our proposed metrics. We also provide an online prediction mechanism based on query utility for deciding whether to personalize a query in UPS. Extensive experiments demonstrate the efficiency and effectiveness of our framework.

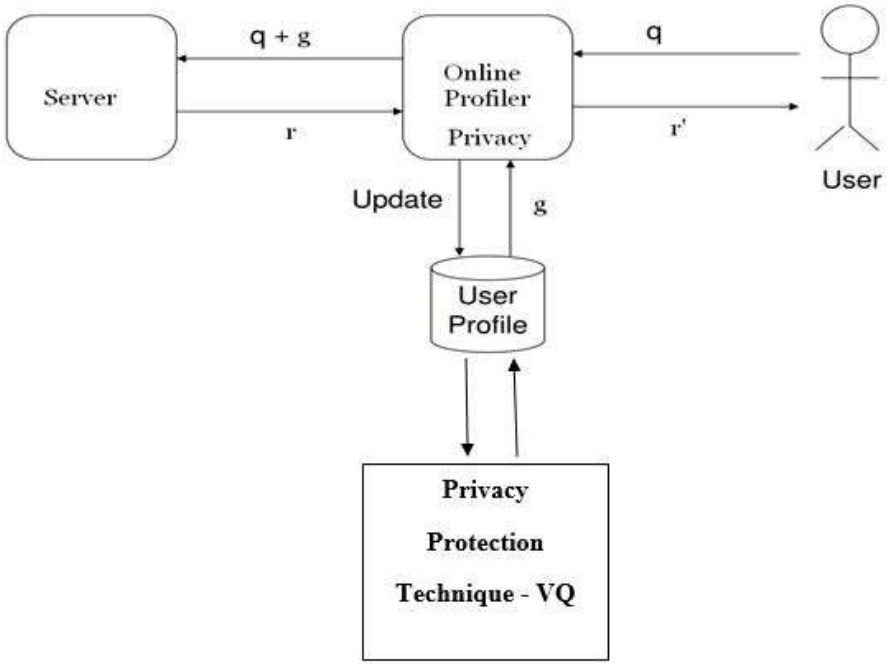
Generating user profile

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

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

Online decision

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



Enhanced Privacy Protection Architecture

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

Algorithm of proposed design

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

The benefits of making the above runtime decision are, it enhances the stability of the search quality and it avoids the unnecessary exposure of the user profile.Therefore, GreedyIL is expected to significantly outperform GreedyDP. The steps for GreedyIL algorithm are

Step 1: If G' is a profile obtained by applying a prune-leaf operation on G, then $DP(q; G) \geq DP(q, G')$.

Step 2: Specifically, each candidate operator in the queue is a tuple like op = (t, IL (t, G_i)), where t is the leaf to be pruned by op and IL (t, G_i), indicates the IL incurred by pruning t from G_i.

Step 3: The iterative process can terminate whenever ϑ-risk is satisfied.

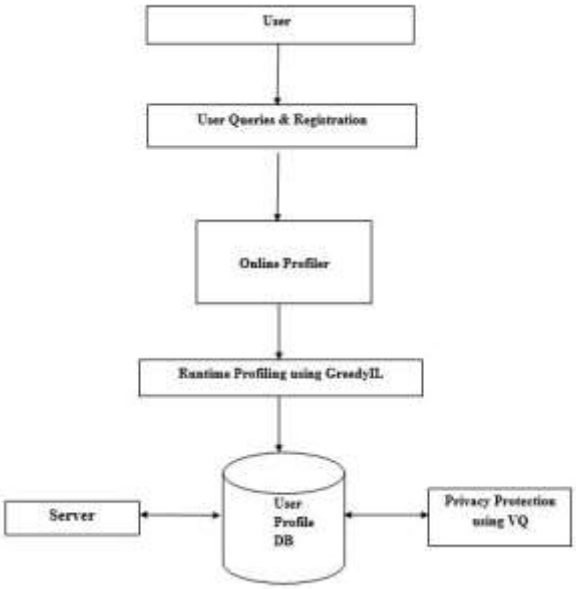
Step 4: The second term (TS(q, G) remains unchanged for any pruning operations until a single leaf is left (in such case the only choice for pruning is the single leaf itself).

Step 5: In C1, t is a node with no siblings, and In C2, t is a node with siblings. The case C1 is easy to handle. However, the evaluation of IL in case C2 requires introducing a shadow sibling of t.

Step 6: Each time if we attempt to prune t, we actually merge t into shadow to obtain a new shadow leaf shadow0, together with the preference of t,

Step 7: Prune-leaf only operates on a single topic t. Thus, it does not impact the IL of other candidate operators in Q. While in case C2, pruning t incurs re-computation of the preference values of its sibling nodes.

Step 8: Once a leaf topic t is pruned, only the candidate operators pruning t's sibling topics need to be updated in Q. In general, GreedyIL traces the information loss instead of the discriminating power. This saves a lot of computational cost.



System Flow Diagram For Proposed Design

Privacy protection technique

The encoding and decoding process of the cryptography method is illustrated below.

Quantization is the procedure of constraining something from a relatively large or continuous set of values (such as the real numbers) to a relatively small discrete set (such as the integers). The discrete cosine transform (DCT) helps separate the text into parts (or spectral sub-bands) of differing importance (with respect to the image's visual quality). The DCT is similar to the discrete Fourier transform but using only real numbers. There are eight standard DCT variants, of which four are common. The most common variant of discrete cosine transform is the type-II DCT, which is often called simply "the DCT"; its inverse, the type-III DCT, is correspondingly often called simply "the inverse DCT" or "the IDCT". Two related transforms are the discrete sine transforms (DST), which is equivalent to a DFT of real and odd functions, and the modified discrete cosine transforms (MDCT), which is based on a DCT of

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

overlapping data. The most powerful and quantization technique used for the cryptography is vector IBC. The IBC uses vector quantization algorithms for reducing the transmission bit. Text vector quantization algorithm includes four stages:

- Vector formation,
- Training Set Selection,
- Codebook Generation and
- Quantization.

The first step is to divide the input into set of vectors. The Subset of vectors in the set is later chosen as a training sequence. The codebook of code words is obtained by an iterative clustering algorithm. Finally, in quantizing an input vector, closest code words in the codebook is determined and corresponding label of this code word is transmitted. In this process, data compression is achieved because address transmission requires fewer bits than transmitting vector itself. The concept of data quantization is extended from scalar to vector data of arbitrary dimension. Instead of output levels, vector quantization employs a set of representation vectors (for one dimensional case) or matrices (for two dimensional cases). Set is defined as —codebook and entries as —code words. Vector quantization has been found to be an efficient coding technique due to its inherent ability to exploit the high correlation between the neighboring pixels

JPEG technique divides the input image into non-overlapping blocks of 8x8 pixels and uses the DCT transformation. For each quantized DCT block, the least two-significant bits (2-LSBs) of each middle frequency coefficient are modified to embed two secret bits. Using gray-level cover images, we transformed (DCT) non-overlapping blocks of 16x16 pixels instead of non-overlapping blocks of 8x8 pixels. The transformed DCT coefficients were quantized by a modified 16x16 quantization table. Then, the secret data is embedded within the middle frequency coefficients

8 X 8 quantization table

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Dividing this quantization table and by 2, a new quantization table is obtained, like below.

The scaled quantization table

8	6	5	8	12	20	26	31
6	6	7	10	13	29	30	28
7	7	8	12	20	29	35	28
7	9	11	15	26	44	40	31
9	11	19	28	34	55	52	39
12	18	28	32	41	52	57	46
25	32	39	44	52	61	60	51
36	46	48	49	56	50	52	50

Using this new quantization table generates reconstructed images almost identical to the source image. The modified version of (Table II), has been used within Chang et al. method. 8x8 quantization tables apart, there are no samples for larger quantization tables in the JPEG standard

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

Modified quantization table

8	6	5	8	1	1	1	1
6	6	7	1	1	1	1	28
7	7	1	1	1	1	35	28
7	1	1	1	1	44	40	31
1	1	1	1	34	55	52	39
1	1	1	32	41	52	57	46
1	1	39	44	52	61	60	51
1	46	48	49	56	50	52	50

Advantages in proposed design

- It enhances the stability of the search quality
- Improves the privacy protection against different type of attacks
- It avoids the unnecessary exposure of the user profile
- It provides runtime profiling

Data Sets

High dimensional data are characterized by few dozen to many thousands of dimensions and any dataset representable under a relational model is chosen as a High Dimensional Dataset. According to that the following six different datasets were used, it is worth noting that the 20NG, Sports, Health, Society, and Local News.

The Category of Dataset

Category	No. of User Profiles
20NG	412
Sports	300
Health	669
Society	442
Local News	254

Performance Evaluation

The following performance parameters are commonly used in privacy protection technique evaluation. The existing approach is compared with proposed approach using these evaluation parameters. The system is evaluated in terms of Precision, Recall, F-measure, Computational Delay and Distortion.

Results and discussions

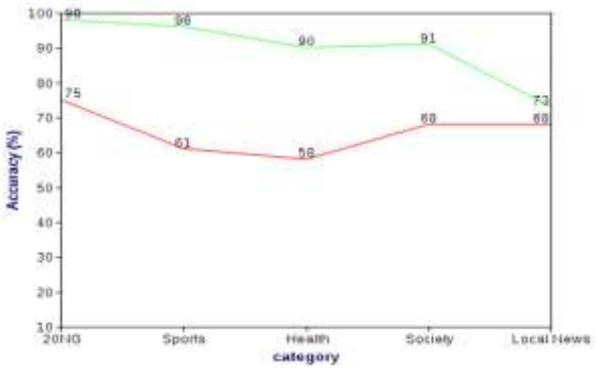
Precision

It is a measure of correctly predicted documents by the system among all the predicted documents. It is defined as the number of relevant documents retrieved by a search divided by the total number of documents retrieved by that search.

precision= number of correct results/ number of all returned results

Precision Comparative

Categories	No. of User Profiles	Precision	
		Existing	Proposed
20NG	412	75%	98%
Sports	300	61%	96%
Health	669	58%	90%
Society	442	68%	91%
Local News	254	68%	73%



Evaluation of Precision using GreedyIL Algorithm

The proposed approach accuracy level is high when compared with the existing one.

Recall

Recall is a measure of correctly predicted documents by the system among the positive documents. Recall is defined as the number of relevant documents retrieved

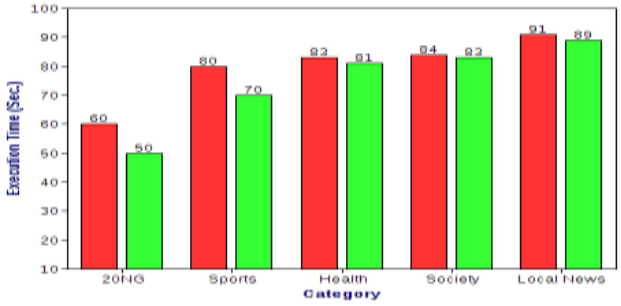
ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

by a search divided by the total number of existing relevant documents.

recall= number of correct results/total number of actual results

Recall Comparative

Categories	No. of User Profiles	Recall	
		Existing	Proposed
20NG	412	60 Sec	50 Sec
Sports	300	80 Sec	70 Sec
Health	669	83 Sec	81 Sec
Society	442	84 Sec	83 Sec
Local News	254	91 Sec	89 Sec



Evaluation of Recall using GreedyIL Algorithm

The proposed approach takes less time when compared with existing design.

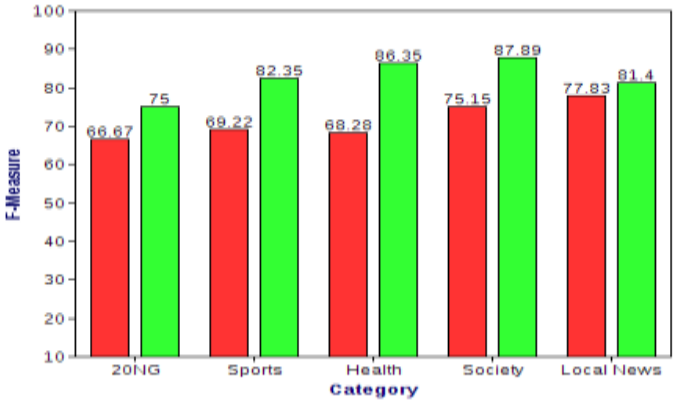
Frequency-Measure

F-measure combines precision and recall and is the harmonic mean of precision and recall.

F-measure=2*(precision*recall/precision+recall)

F-Measure Comparative

Categories	No. of User Profiles	F-Measure	
		Existing	Proposed
20NG	412	66.67	75
Sports	300	69.22	82.35
Health	669	68.28	86.35
Society	442	75.15	87.89
Local News	254	77.83	81.40



Evaluation of F-Measure using GreedyIL

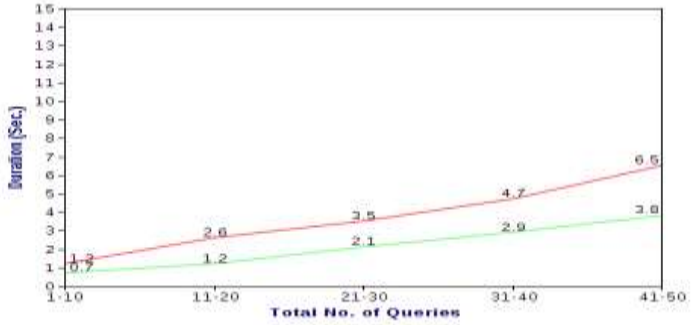
Frequency measures are very helpful in evaluating the performance of both frequent and rare categories.

Computational Delay

It represents the accessing time or speed of user profiles in the database.

Computational Delay Comparative

Categories	No. of User Profiles	Computational Delay	
		Existing	Proposed
20NG	412	1.2	0.7
Sports	300	2.6	1.2
Health	669	3.5	2.1
Society	442	4.7	2.9
Local News	254	6.5	3.8



Evaluation of Computational Delay using GreedyIL

The proposed approach takes less computational time for accessing the queries when compared with existing design.

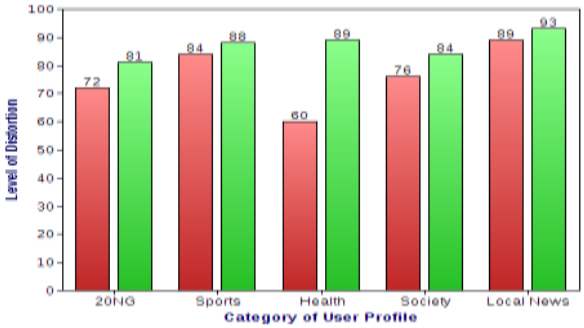
Distortion

The Distortion is used to measure the level between original dataset and changed dataset.

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

Distortion Comparative

Categories	No. of User Profiles	Distortion Level (%)	
		Existing	Proposed
20NG	412	72	81
Sports	300	84	88
Health	669	60	89
Society	442	76	84
Local News	254	89	93



Evaluation of Distortion using GreedyIL

The proposed level of distortion is high when compared with existing design.

Conclusion

The remarkable development of information on the Web has forced new challenges for the construction of effective search engines. The proposed work provides information on user customizable privacy preserving search framework-UPS for Personalized Web Search. 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. Another important conclusion we revealed in this proposed work is that personalization does not work equally well under various situations. The click entropy is used to measure variation in information needs of users under a query. Experimental results showed that personalized Web search yields significant improvements over generic Web search for queries with a high click entropy. For the queries with a low click entropy, personalization methods performed similarly or even worse than generic search. As personalized search had different effectiveness for different kinds of queries, we argued that queries should not be handled in the same manner with regard to personalization. The proposed click entropy can be used as a simple measurement on whether a query should be personalized.

References

1. Lidan Shou, He Bai, Ke Chen, and Gang Chen, "Supporting Privacy Protection in Personalized Web Search," vol . 26, no. 2, Feb 2014
2. J.S. Breese, D. Heckerman, and C.M. Kadie, "Empirical Analysis of Predictive Algorithms for Collaborative Filtering," Proc. 14th Conf. Uncertainty in Artificial Intelligence (UAI), pp. 43-52, 1998
3. 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.
4. J. Conrath, "Semantic Similarity based on Corpus Statistics and Lexical Taxonomy," Proc. Int'l Conf. Research Computational Linguistics (ROCLING X), 1997.
5. J. Castellı´-Roca, A. Viejo, and J. Herrera-Joancomartı´, "Preserving User's Privacy in Web Search Engines," Computer Comm., vol. 32, no. 13/14, pp. 1541-1551, 2009.
6. P.A. Chirita, W. Nejdl, R. Paiu, and C. Kohlschu¨ tter, "Using ODP Metadata to Personalize Search," Proc. 28th Ann. Int'l ACM SIGIR Conf. Research and Development Information Retrieval (SIGIR), 2005.
7. Dou, Zhicheng, Ruihua Song, and Ji-Rong Wen. "A large-scale evaluation and analysis of personalized search strategies." Proceedings of the 16th international conference on World Wide Web. ACM, 2007.
8. Z. Dou, R. Song, and J.-R. Wen, "A Large-Scale Evaluation and Analysis of Personalized Search Strategies," Proc. Int'l Conf. World Wide Web (WWW), pp. 581-590, 2007.
9. O. Etzioni. The world wide web: Quagmire or gold mine. Communications of the ACM, 39(11):65–68,1996.
10. S. Gauch, J. Chaffee, A. Pretschner, Ontology-Based User Profiles for Search and Browsing, User Modeling and User-Adapted Interaction: The Journal of Personalization Research, Special Issue on User Modeling for Web and Hypermedia Information Retrieval, vol. , (2003).
11. K. Hafner, Researchers Yearn to Use AOL Logs, but They Hesitate, New York Times, Aug. 2006.
12. A.Krause and E. Horvitz, "A Utility-Theoretic Approach to Privacy in Online Services," J. Artificial Intelligence Research, vol. 39, pp. 633-662, 2010.

ISSN: 2277-9655
Scientific Journal Impact Factor: 3.449
(ISRA), Impact Factor: 2.114

13. R. Kosala, H. Blockeel "Web mining research: A survey," ACM SIGKDD Explorations, Vol. 2 No. 1, pp. 1-15, June 2000
14. X. Shen, B. Tan, and C. Zhai, "Context-Sensitive Information Retrieval Using Implicit Feedback," Proc. 28th Ann. Int'l ACM SIGIR Conf. Research and Development Information Retrieval (SIGIR),2005.
15. B. Tan, X. Shen, and C. Zhai, "Mining Long-Term Search History to Improve Search Accuracy," Proc. ACM SIGKDD Int'l Conf.Knowledge Discovery and Data Mining (KDD), 2006.