SEARCH RESULT CLUSTERING FOR WEB PERSONALIZATION

Kavita D. Satokar, A. R. Khare
Researcher, Mtech (IT), Department of Information Tech., B.V.D.U.C.O.E., Pune, India
Assistant Professor, Department of Information Tech., B.V.D.U.C.O.E., Pune, India.

ABSTRACT

The main problem faced by the users of web search today is the quality and the amount of the results they get back. The results frustrate a user and consume his precious time. Existing search engines perform keyword based searches without taking into account the user intent and semantics of the user query. Hence to improve searching in the WWW, a new personalized search index provides a conceptual relation between the search keywords and the pages, which matches the user's information need. The proposed approach aims to mine a reduced set of effective search result for enhancing the searching experience. In this project, we propose and build a personalized 2D web search model. We store and maintain user's long-term dynamic profile based on user search and use it to personalize. We use ontology at client side to solve the cold start problem and expand the query and generate clusters of similar results. We store client's profile as a weighted ontology tree. We use web search results from an existing search engine and re-rank them based on client's profile.

KEYWORD: clustering, web personalization, ontology, web mining

I. INTRODUCTION

Web search is difficult because it is hard for users to construct queries that are both sufficiently descriptive and sufficiently discriminating to find just the web pages that are relevant to the user's search goal. Ambiguous queries lead to search result sets containing distinct page groups that meet different user search goals. To filter out irrelevant results the users must refine their search by modifying the query .Users must understand the result set to refine queries effectively; but this is time consuming, if the result set is unorganized. Web personalization using web search result clustering is one approach for assisting users to both comprehend the result set and to refine the query. According to Eirinaki and Vazirgiannis [3] personalization is defined as follows: Web site personalization can be defined as the process of customizing the content and structure of a Web site to the specific and individual needs of each user taking advantage of the user's navigational behaviour.Web page clustering identifies semantically meaningful groups of web pages and presents these to the user as clusters. The clusters provide an overview of the contents of the result set and when a cluster is selected the result set is refined to just the relevant pages in that cluster. An ontology is a model of the world, represented as a tangled tree of linked concepts. Concepts are language-independent abstract entities. They are expressed in this ontology using English words and phrases only as a simplifying convention. Semantic ontology is to improve automated text processing by providing language-independent, meaning-based representations of concepts in the world. The ontology shows how concepts are related and their properties.The objective of a web personalization system is to provide users with the information they want or need, without expecting from them to ask for it explicitly.
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

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

Any system providing personalization services will need to store some information about the user in order to achieve its goal. The simplest way to construct a profile is to collect users' preferences explicitly, by asking them to submit the necessary information manually before any personalization can be provided. However, studies like [6] show that users are generally not willing to spend extra time and effort on specifying their intentions especially when the benefits may not be immediately obvious. There are also often concerns about privacy, and users might not be very comfortable supplying personal information to search servers.

Section I indicates the work done in the field of search personalization. Section II describes the proposed search personalization system. Section III describes the proposed OntoPersonalization Ranking algorithm. The experimental results and conclusion are explained in later sections

II. RELATED WORK

Hearst and Pedersen [2] showed that relevant documents tend to be more similar to each other, thus the clustering of similar search results helps users find relevant results. Several previous works [8][9][2][5][4] are conducted to develop effective and efficient clustering technology for search result organization. In addition, Vivisimo [7] is a real demonstration of this technique. Lee and Bordin defines a class of personalized search algorithms called "local-cluster" algorithms that compute each page's ranking with respect to each cluster containing the page rather than with respect to every cluster. In particular, they propose a specific local-cluster algorithm by extending the approach taken by Achlioptas et al. [10]. They proposed local-cluster algorithm considers linkage structure and content generation of cluster structures to produce a ranking of the underlying clusters with respect to a user's given search query and preference. The rank of each document is then obtained through the relation of the given document with respect to its relevant clusters and the respective preference of these clusters. Zamir and Etzioni [8][9] presented a Suffix Tree Clustering (STC) which first identifies sets of documents that share common phrases, and then create clusters according to these phrases. Our candidate phrase extraction process is similar to STC but we further calculate several important properties to identify salient phrases, and utilize learning methods to rank these salient phrases. Some topic finding [1][3] or text trend analysis [9] works are also related to our method. The difference is that we are given titles and short snippets rather than whole documents.

Motivated by Lee and Bordin's local cluster algorithm we propose a cluster based probability algorithm. Our algorithm considers cluster probability, user choice obtained through local webcluster database and defined ontology to produce a ranking of the underlying clusters with respect to a user's given search query and preference. The rank of each document is then obtained through the relation of the given document with respect to its relevant clusters and the respective preference of these clusters.

III. PROPOSED SYSTEM

In any Information Retrieval model the important challenge is to present the results which the user is expecting for his query. Efficiency is a challenge in this that has been addressed very well so far. Current web search engines serve all users, independent of the special needs of any individual user. Personalization of web search is intended to carry out retrieval for each user incorporating his/her interests. Even though there exist some personalization models that facilitate personalization to some extent they fail in cases where the results are totally biased towards a dominant keyword in the search query. Generally, the user doesn't want to go beyond two pages of results. And in most cases the results relevant to the dominant keywords fill up the first few pages making the user unsatisfied.

We propose a client side personalization model that would effectively overcome the above stated problems. The system uses a middleware approach. We build entity search capabilities on top of an existing search-engine such as Goggle by "wrapping" the original engine. The middleware would take

a user query, use the search engine API to retrieve top K web pages most relevant to the user query, and then cluster those web pages based on their associations to real people. The architecture is a pipeline that receives the input query, obtains search results from a search engine, filters the results applying a clustering algorithm and then gets the clusters. The steps of overall approach are illustrated in Fig 1

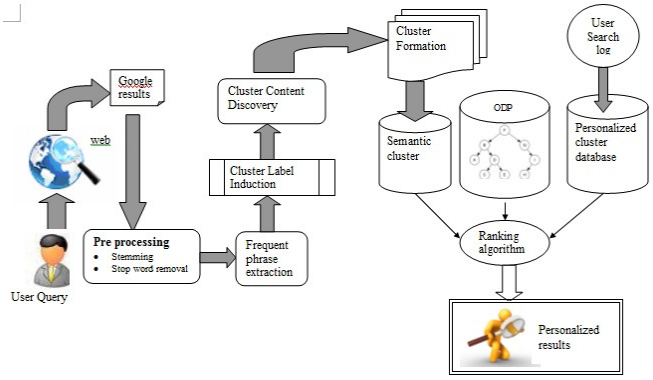

Fig 1: System Diagram

The system is divided into three major sections

1. Search Result Fetching

We first get the WebPages of search result lists returned by a web search engine. We have extracted results from yahoo, google and msn. So the first search is the conventional meta-search based on these keywords. These WebPages are analysed by an HTML parser and the result items are extracted. Generally, there are only titles and query-dependent snippets available in each result item. We assume these contents are informative enough because most search engines are well designed to facilitate users' relevance judgment only by the title and snippet, thus it is able to present the most relevant contents for a given query. Each extracted phrase is in fact the name of a candidate cluster, which corresponds to a set of documents that contain the phrase.

2. Cluster formation:

The system first identifies meaningful cluster labels and only then assigns search results to these labels to build proper clusters. The algorithm consists of five phases. Phase one is pre-processing of the input snippets, which includes tokenization, stemming and stop-word marking. Phase two identifies words and sequences of words frequently appearing in the input snippets. In phase three, a matrix factorization is used to induce cluster labels. Phase four snippets are assigned to each of these labels to form proper clusters. The assignment is based on the Vector Space Model (VSM) and the cosine similarity between vectors representing the label and the snippets. Finally, phase five is post processing, which includes cluster merging and pruning. The algorithm is as follows:

/ Phase 1: Pre processing */**
for each document
{
do text filtering;
identify the document's language;
apply stemming;
mark stop words;
}
/ Phase 2: Feature extraction */**
discover frequent terms and phrases;
/ Phase 3: Cluster label induction */**
use LSI to discover abstract concepts;
for each abstract concept

```
{
find best-matching phrase;
}
prune similar cluster labels;
/** Phase 3: Cluster label induction */
use LSI to discover abstract concepts;
for each abstract concept
{
find best-matching phrase;
}
prune similar cluster labels;
/** Phase 4: Cluster content discovery */
for each cluster label
{
use VSM to determine the cluster contents;
}
/** Phase 5: Final cluster formation */
calculate cluster scores;
apply cluster merging;
```

3. Cluster ranking:

Finally, clusters are sorted for display based on their score, calculated using the following simple formula: C_{score} = label score × ||C||, where ||C|| is the number of documents assigned to cluster C. The scoring function, although simple, prefers well-described and relatively large groups over smaller, possibly noisy ones.

As we retrieved the original ranked list of search result $R=\{r(di|q)\}$, where q is current query, di is a document, and r is some (unknown) function which calculates the probability that di is relevant to q. Traditional clustering techniques attempt to find a set of topic-coherent clusters C according to query q. Each cluster is associated with a new document list, according to the probability that di is relevant to both q and current cluster:

$C = \{Rj\}$, where $Rj = \{r(di|q, Rj)\}$ (1)

In contrast, our method seeks to find a *ranked* list of clusters C', with each cluster associated with a cluster name as well as a new ranked list of documents:

$C' = \{r'(ck, Rk|q)\}$, where $Rk = \{r(di|q, ck)\}$ (2)

As shown in Eq. 1 and Eq. 2, we modify the definition of clusters by adding cluster names ck, and emphasize the ranking of them by function r'.

The OntoPersonalization algorithm for ranking clusters is the direct analogy of the SP algorithm where now clusters play the role of pages. That is, we will be interested in the aggregation of links between clusters and the term content of clusters. In order to incorporate the user relevant query-independent web page importance, personalized result ranks, ontology and original web ranks (as an approximation for the real page rank) are aggregated to form the final result ranking. Thereby each result item is assigned a score (indicating probability) corresponding to the number of results ranked below it. Then the total score of a result is a weighted sum of its scores with respect to each ranking, i.e.,

score = (w*0.7)_score1 +(0.3 * w)_score2 +C_score (3)

where score is the final score of the result item, based on which the results are finally re-ranked before being submitted to the user, score1 is the score of the result item within the personalized result set, score2 is the score of the result item within the given ontology result set, and such that the

combination threshold w serves as a personalization control parameter. C_score is the score obtained by Lingo algorithm. By adjusting the value of w, the user controls the personalization level. We have considered a 70-30 ratio for personalization and Ontology tree. The threshold can be set by the user. For instance, setting w to 0 would mean that the user would be presented with the original result ranked on the basis of defined ontology tree, and on the other hand setting w to 1 would mean that the new result set is the same as the personalized set. Our web search personalization system provides a control over threshold value w, thus enabling the user to cancel the personalization at any point of time. The figure 2 indicates the overall ranking process.

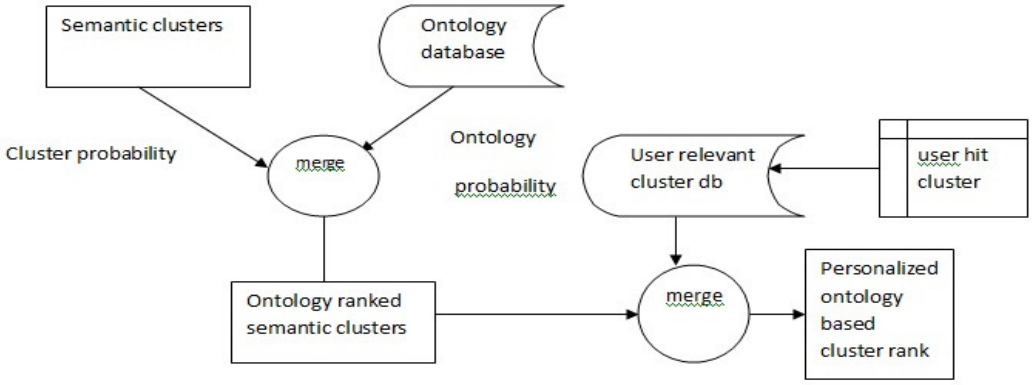

Fig 2: The Cluster Merging Diagram

IV. THE ONTO PERSONALIZATION RANKING ALGORITHM

Our goal is to utilize the user context to personalize search results by re-ranking the results returned from a search engine for a given query Assuming an ontological user profile with interest scores exists and we have a set of search results, Algorithm is utilized to re-rank the search results based on the interest Scores, user choice score and the semantic cluster score. The proposed algorithm is capable of presenting results according to the user desired level of personalization.

The Onto Personalization algorithm works in three steps

1. Ontology Ranking
2. Pure personalization Ranking
3. Final Ranking

The algorithm uses previous clicked cluster data, stored in webcluster database,to create task oriented dynamic profile of user. The ontology for the given keyword is identified and extracted from the ontology database. Thus appropriate weights are added to the cluster depending on given personalization level. Thus using dynamic user profile and ontology cluster list we finally obtain a ranked cluster list which satisfies the user intent is obtained.

Algorithm: Onto Personalization Ranking

Input: Cluster List, Ontology Database, Webcluster Database,User Query

Output: Ranked Output: Ranked Personalized List

Steps:

1. Get the user query.
2. Retrieve the ontology tree node, on, matching the query.
   /*this node acts as parent node*/
3. Get all child nodes, add weight to them.
4. Get user query
5. Retrieve all records where query matches the webcluster database keyword.
6. Add weights to those clusters
7. Get user personalization –Ontology ranking ratio.
8. Multiply the ontology list with the ontology ratio.
9. Multiply personalized list personalization ratio.
10. Merge the ontology and personalization list

11. Match the list with the semantic cluster list obtained from Lingo algorithm.
12. Discard non matching clusters.
13. Re rank the final list.

V. EXPERIMENTAL RESULTS

The figure 3 shows the system interface of our proposed system for the query "computer mouse".

Fig 3: System Interface

The figure 4 shows how the cluster score increases as the user searches the same term repeatedly. Figure 5 indicates the user satisfaction percentage with the offered results. The graph shows the increase in cluster ranking as the system learns more about the intent of user.

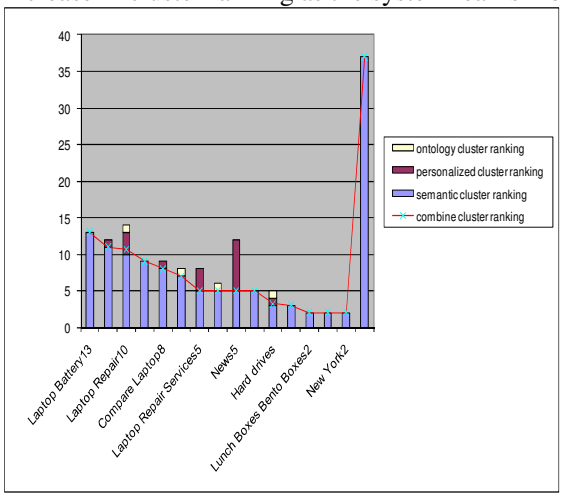

Fig 4: Results showing increasing in cluster ranking depending on user choice

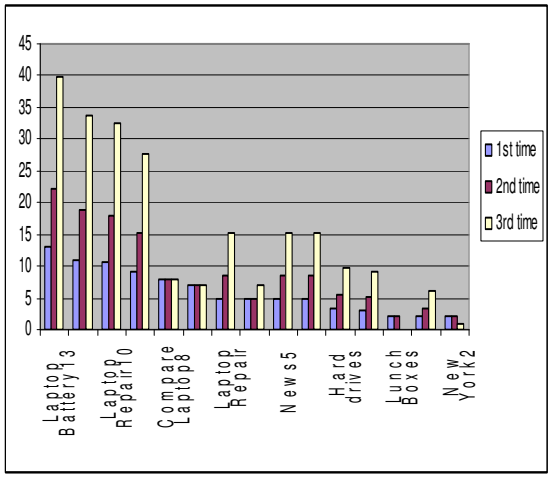

Fig 5:Growth of personalization

VI. CONCLUSION

We have introduced a web mining tool, a personalized, knowledge-driven cluster based search system that helps the user to find web information based on individual preferences. Analysing the currently

available algorithms, we observed that little emphasis was being placed on the quality of thematic groups' description. The aim of this work is to perform personalized search by recording user profile for users from their browsing pattern and to retrieve more relevant and related documents clusters that are semantically related to the given search query. To achieve this, it is essential to know the meaning and domain of the search query. To understand the semantics of the search query, Ontology is developed. Along with the semantic clusters ranking probability, the ontology ranking and personalized cluster probability is taken into account to decide the final ranking of clusters for a given search query. Personalization using such ontologies and semantic can produce better results as compared to the keyword-based searching.

Our system also shows that, while it is possible to improve the efficiency of search through each of the personalization methods discussed above, they infact work best when operated in conjunction with one another, acting as a checks and balances mechanism. When used in conjunction, the inferences truly become more probable, and lead to dramatically better search results.Effecient information gathering without disturbing the privacy of user can still prove a good way to personalize the search results.

REFERENCE

[1] Web Page Personalization based on Weighted Association Rules by R. Forsati M. R. Meybodi,A. Ghari Neiat.Department of computer engineering,Islamic Azad University, North at 2009 International Conference on Electronic Computer Technology . DOI 10.1109/ICECT.2009.104.

[2] Hyperlink Classification: A New Approach to Improve PageRank by Li Cun-he, Lv Ke-qiang 18th International Workshop on Database and Expert Systems Applications 2007 IEEE DOI 10.1109/DEXA.2007.14.

[3] Web Mining for Web Personalization by MAGDALINI EIRINAKI and MICHALIS VAZIRGIANNIS Athens University of Economics and Business ACM Transactions on Internet Technology, Vol. 3, No. 1, February 2003, Pages 1–27.

[4] An Empirical Evaluation on Semantic Search Performance of Keyword-Based and Semantic Search Engines: Google, Yahoo, Msn and Hakia Duygu Tümer1, Mohammad Ahmed Shah2, Yıltan Bitirim1 2009 Fourth International Conference on Internet Monitoring and Protection IEEE.

[5]Web Usage Mining based on Clustering of Browsing Features Chu-Hui Lee Yu-Hsiang Fu Eighth International Conference on Intelligent Systems Design and Applications.

[6]Web Search Personalization with Ontological User Profiles by Ahu Sieg,Robin Burke,CIKM'07, November 6–8, 2007, Lisboa, Portugal.ACM 978-1-59593-803-9/07/0011.

[7]Honghua Dai and Bamshad Mobasher-"Integrating Semantic Knowledge with Web Usage Mining for Personalization"

[8] *Topic Sensitive PageRank* by Haveliwala

[9]Learning Implicit user History Using Ontology and Search History for Personalization by Mariam Daoud, Lynda Tamine, Mohand Boughanem and Bilal Chebaro

[10] D. Achilioptas, A. Fiat, A. R. Karlin, and F. McSherry. Web search via hub synthesis. In FOCS, pages 500–509. ACM, 2001.

[11] Clustering hyperlinks for topic extraction: an exploratory analysis by Sara Elena Gaza Villarreal,Tecnol´ogico de Monterrey,Eugenio Garza Sada, This paper appears in 2009 Eighth Mexican International Conference on Artificial Intelligence 2009 IEEE DOI 10.1109/MICAI.2009.20

[12] Adaptive User Profiling for Personalized Information Retrieval by Hochul Jeon, Taehwan Kim, Joongmin Choi This paper appears in Third 2008 International Conference on Convergence and Hybrid Information Technology 2008 IEEE ,DOI 10.1109/ICCIT.2008.111

[13] Web Search Personalization by User Profiling by Mangesh Bedekar, Dr. Bharat Deshpande, Ramprasad Joshi This paper appears in First International Conference on Emerging Trends in Engineering and Technology 2008 IEEE DOI 10.1109/ICETET.2008.70

[14] An Individual WEB Search Framework Based on User Profile and Clustering Analysis by Jie Yuan, Xinzhong Zhu*, Jianmin Zhao, Huiying Xu at International journal of Computer Sciences IEEE vol 2008

[15] Web Search with Personalization and Knowledge by George T. Wang, F. Xie F. Tsunoda, H. Maezawa,,Akira K. OnomaThis paper appears in: Proceedings of the IEEE Fourth International Symposium on Multimedia Software Engineering (MSE'02)

Author Biography

K. D. Satokar (Alias K. P. Moholkar) is a Research Scholar. She is working as an Assistant Professor in Computer Engineering Department Of Rajarshi Shahu College Of Engineering ,Pune, India. She has total 10 years teaching experience in the department of Computer Engineering. She specializes in subjects like database and web mining and Artificial Intelligence. The present work is a part of her ongoing research. She is working on this topic for last 2 years.

Akhil Khare is working as an Associate Professor in BVDU COE pune, India. He was awarded his M. Tech (IT) Degree from Government Engg. College, BHOPAL in 2005. His areas of interest are Computer Network, Software Engineering, Multimedia System and Data Processing. He has Eight years' experience in Teaching and Research. He has published more then 35 research papers in journals and conferences. He has also guided 10 postgraduate scholars.