Domain Knowledge Driven Personalized Web Search

[1]Prof.Vishal R.Shinde, [2]Kedar P. Pandhare, [3]Raj D. Patil, [4]Vaibhav V. Patil
[1]*Asst. Professor,* [2,3,4]*BE Student Computer Engg. Dept. SSJCET, Asangaon, India.*
[1]*mailme.vishalshinde@gmail.com,* [2]*pandhare.kedar@yahoo.com,* [3]*rd7.patil@gmail.com,*
[4]*vaibhav.v.patil91@gmail.com.*

Abstract - The generic search engines existing today are using 'search as you type' technique while searching the information in order to retrieve information faster as compared to keyword search. But generic search engine fails to distinguish different users. Each user is unique and has got his/her unique interests. The generic search engine shows same results to different users without being bothered of their needs/interests. Most of the personalized search engines retrieve the relevant information but use only the keyword search technique. The project proposes domain knowledge driven personalized web search by using the typing technique in order to retrieve pertinent information and compare the results with that of keyword search.

Keywords — Browser history based search, Efficient Web Search by cache data, Domain knowledge, Personalized web search, User modeling algorithm, Page Rank.

I. INTRODUCTION

Web search engine is a software based system that is designed to search for information on the World Wide Web. The results obtained after searching are generally presented in a line of results often referred to as search engine result pages. The information may be a mixture of web pages, images, and other types of files. Some search engines perform mining of the data that is available in databases or open directories. Unlike most of the web directories, which are maintained only by human editors, search engines also perform the task of maintaining real-time information by running an algorithm on a web crawler. Web search engines perform the work by storing information about several web pages, and later retrieve them from the HTML mark-up of the pages. These pages are retrieved by a Web crawler which follows every link on the site. Search engine then analyses the contents of each web page and determines how indexing should be performed(for example, words can be extracted from the titles, large page content, headings, or other special fields called meta tags). Data regarding web pages are stored in an index database for the use in later queries. A query from a user can be a single word or group of words. The index helps find information relating to the query as quickly as possible. This cached page always holds the actual search text since it is the one that was actually indexed, hence it can be very fruitful when the content of the current page has been updated and the search terms are no longer in it. This paper proposes architecture for constructing search engine using domain knowledge and user history.

II. LITERATURE SURVEY

Let's take analysis of different developed search engines methodologies for efficient search results and the proposed method for crucial web page results. Different search engine approaches are applicable for efficient prediction of web page results. Some of the search engines have their own working model with special features. Google has features like android specific application where as Microsoft Bing has application specific to windows platform and yahoo is lagging far behind them, but their main focus is at services like Yahoo Mail, Yahoo answer. Every search engine has its own advantages and disadvantages.

Domain Knowledge Driven Personalized Web Search Engine allows user to traverse easily by suggesting his domain related interested pages at runtime with help of its browsing history [1]. Some popular search engines comparison based on crucial points to cover variety of features provided by them is gathered and it is shown in following table,

	Google	***Bing***	***Yahoo***	***Pws***
Users	1.17 billion (approx.)	Over 100 million	Over 800 million	beta stage
Shares	707.71USD6.80 (0.97%)	33% of total gross.	30.44USD0.39 (1.31%)	nil
Technic	mobile apps & softwares	Limited only to search domains	Search engines & emails	Restricted only to search engines.
Pros	Faster search results as compared to other	known for good add-ons, including travel and local results	email, instant messaging, social networks and SMS	User friendly & more reliable than other web browser
Cons	no inventory level guarantees	Bing tends to lag in optimization	You cannot label messages freely	No such con detected till date.
Features	Search engines, maps, cloud storage, mails, social networking	Account privacy, Microsoft edge.	Bulk mail storages (upto 1tb data)	Hits calculator for specific user domain data basis
Services	Play store, google plus, android app	MSDN Microsoft, Spatial data service	Yahoo help central.	not restricted to specific results
Achievements	Top notch in search engines family. Universally affiliated.	No big achievements yet.	Revolutionary mail service till date.	Successfully managed the domain based search
Algorithm	PageRank algorithm only.	Anchor text links.	Yahoo search rank algorithm.	Proposed algorithm
Result	Restricted only to a particular domain.	Not enough data provided in singular keywords	Limits search results as per user location.	Results are precise and based on previously search data items.

Table I: Comparative Study

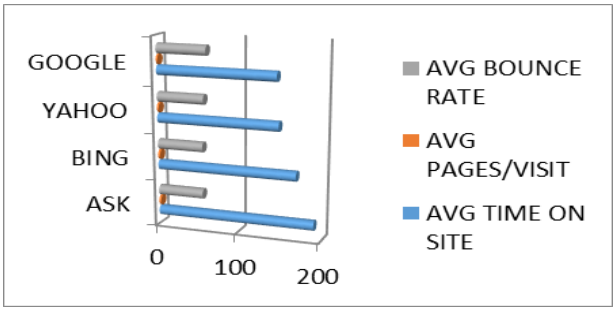

Fig 1: Statistical Graph

III. AIM AND OBJECTIVE

Web search engines commonly provide search results without considering the user interests or context. It proposes a personalized search methodology that can easily extend a conventional search engine on the client side as well as on the server side. The mapping framework automatically maps a set of known user interests onto a group of categories in the database and takes advantage of manually edited data that is available in database for training the text classifiers, thereby categorizing and personalizing search results according to user interests.

Objective

1. More accurate the data more accurate the result.
2. Objective in proposed is to provide efficient algorithm for page hits calculation
3. Objective here is not only to have correct prediction for web pages but to make algorithm generalized
4. Correctly detected bounced results & false entries rectification will give us the correct prediction while searching.

IV. STAGES IN PROPOSED SYSTEM

Proposed architecture can be shown as follows:

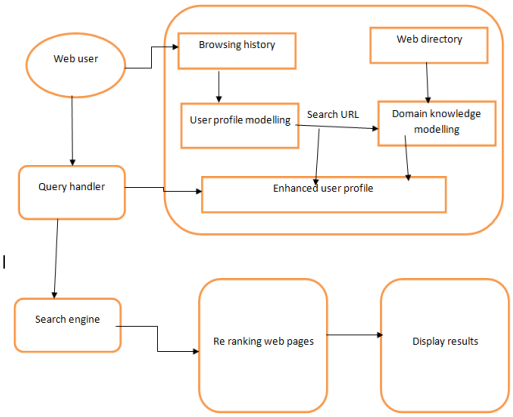

Fig 2: Proposed System

a) Personalized Web Search Module:

Personalized web search takes individual's interest into consideration and enhances the usual web search by suggesting the relevant pages pertaining to his/her interest [5]. A simple and efficient model is proposed which guarantees good suggestions as well as the promises for effective and relevant information retrieval. In addition to this, there is proposed framework implemented for suggesting relevant web pages to the user.

b) User Modeling Module

The proposed system considers user's profile and domain knowledge in order to perform personalized web search. Using the domain knowledge, the system stores information about different domain/categories. Information that is obtained from User Profile is then classified into these specified categories. The learning agent learns user's choice automatically by analyzing the user navigation/browsing history, and thereby creates/updates enhanced User Profile conditioning to the user's most recent choice. Once the query is input by the user, the system provides good suggestions for personalized web search based on enhanced user profile.[2] Further the model makes good use of the advantages of popular search engines, as it can easily re-rank the results obtained by the search engine based on the enhanced user profile.

c) Domain Knowledge Modeling Module:

Domain knowledge is defined as the background knowledge that is used to enhance the user profile. The source which is used for preparing Domain Knowledge is 'Domains' [12]. For preparing Domain Knowledge, first the web pages are crawled from some specified Domain, where each category of the domain is represented by collection of URL's present in that category.

d) Enhanced User Profile Module:

Using the information of user browsing history and domain knowledge, an Enhanced User Profile is created. Once the Enhanced User Profile is created, the user query is taken and the relevant web pages are suggested with respect to the query. In the Experiment, User Profile is used as a base case for suggesting the relevant pages and the results are compared with the pages suggested from Enhanced User Profile. For each query, suggestions upto 20 relevant documents from User Profile are provided and for the same query suggestions upto 20 relevant documents from Enhanced User Profile are provided. In order to compare the efficiency of the result, the similarity of suggested documents is compared with the user query.

V. ALGORITHMS

5.1 Page rank calculation

The page rank algorithm is used to describe the order in which the web pages would be displayed to the user once he/she starts surfing. The page rank would be calculated based upon the most recent webpages that are visited. For every webpage that the user visits, there would be a session id maintained with every web page that the user visits. The session id would serve as a tracker to the various web pages that the user visits. Also hits will be calculated for the webpage. The formula for calculating hits on a Webpage is as follows,

$hits = 0;

$hits = $hits + 1

5.2 Mathematics of Page Rank

Page rank is calculated based upon the hits that are calculated when the user visits the desired webpage. If a particular user visits a webpage, then at the server side there would be a record that is maintained for the particular session during which the web page was visited. Thus as the user starts surfing for different web pages, the session id would be updated for all the web pages that have been visited & so would the hits be incremented. Once the user has completed surfing, the webpages are ranked according to the maximum number of hits that the webpage has recieved.If the user searches for "Google" website and visits the website 15 times, then there would be 15 hits recorded at server end. As the user starts typing next time in the search box, he/she would be suggested with the "Google" site first because of the ranking of the pages according to the maximum hits. PageRank is calculated as follows:

```
$query = select * from url_add where uid='$s' ORDER BY hits DESC
If (is_array ($query))
{
Foreach ($query as $row)
{
   Echo $row ['url'];
}
}
```

5.3 Probabilistic models

This section presents two probabilistic models and inference algorithms for computing the probability that a particular document D is relevant to user U for the query Q. These are called generative models because they describe the procedure by which a user decides whether a particular document is relevant to a particular query. A document about the topic Td is assumed pertinent to a user looking for subject Tu if both:

1. Topic Td satisfies a user with information needed by Tu.
2. If the document's topic matches that of the search intent, then the document is considered relevant to the query.

A) Model 1 (No background model)

This framework is transposable, with many different data sources able to feed into this distribution. The objective is to gain access to the searches that the user has previously surfed and assign it to the session of the user. This would help to know what the user desires to search and based upon the recent web page visits, the related web pages can be easily determined.

Algorithm 1(search results)

```
$result = $_REQUEST ['search'];
$query = select * from url_add where url ='$result'
If (is_array ($query))
{
Foreach ($query as $row)
{
   Echo $domain = $row ['domain'];
}
}
```

B) Model 2 (Background Model)

The background model refers to the data which is obtained when the user is searching for a particular query. For example: if a user searches for a 'Tutorial' webpage like "w3schools.com", then the work of the background model is to gather all the information related to the query that the user is searching. This background data would enable the user to efficiently search different web pages which are relevant to the recent subject that was searched. For every query, the background model would analyse the query that the user is searching and then would gather more information from the database/server and display it to the user. The background model would reduce the task of the user to go back and search for the particular query related to the one which he previously searched. It helps user by suggesting his favourable search domains and allows concurrent tracking of his interest for future efficiency in results

Algorithm 2(Domain related suggestions)

```
$query1 = select * from url_add where domain  ='$domain'
If (is_array ($query1))
{   Foreach ($query1 as $row)
{   Echo $row ['url'];
}  }
```


VI. EXPECTED OUTPUT

We first allow the user to login in to the search portal where he/she after registration would be allowed to search for the desired web content. Once the user searches for the related web content, he would be provided with the related web pages that share same domain as the web page he has searched before. If the user searches for " English songs" which is included in "Music" domain, then he/she would be suggested with related searches like "90'sRock","Hindi Songs",etc that belong to the same domain "Music" Once the user starts navigating to different web pages, we would perform web traversal of the web pages that the user searches along. This would allow the user to easily keep track of all the webpages that he/she has navigated .There would be special section for viewing the web history of the user.

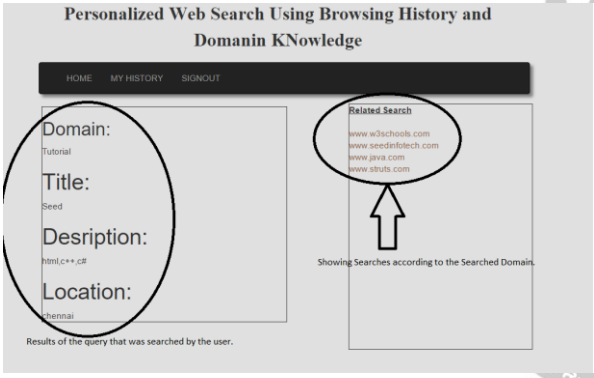

Fig 3: Expected output

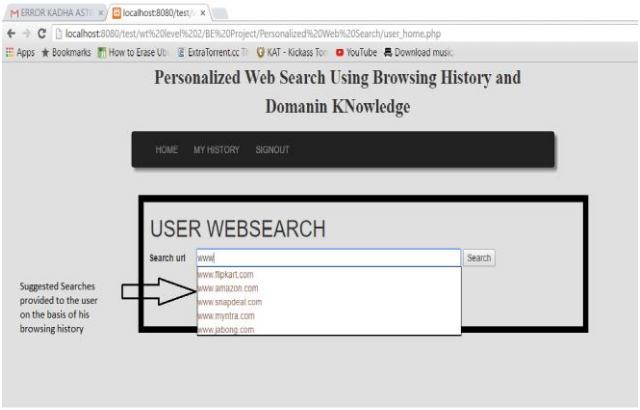

Fig 4: Suggestions based on browsing history

VII. CONCLUSION

The primary goal is to facilitate efficient and improvised web search experience to the user. With the help of various new technologies, it can be easy to enhance the web experience to the user. The user would be able to navigate quickly to related web sites which would indeed improvise the search experience. The concept of personalizing the web search according to user would reduce the burden of navigating to different web sites by typing the name of the website in the search bar. The user would have the freedom of accessing all the related web pages and this would pose as a turning point in the user's web experience.

REFERENCES

[1] Ji-Rong Wen, Zhicheng Dou, Ruihus Song, "*Personalized Web Search",* Microsoft Research Asia, Beijing, China, 2009.
[2] P.A. Chirita, C. Firan, and W. Nejdl, "*Summarizing Local Context to Personalize Global Web Search*", Proc. ACM Int'l Conf. Infor. and Knowledge Management (CIKM), 2006.
[3] G. Atardi, A. Gulli, and F. Sebastiani. Theseus: *categorization by context*. In WWW8, 1999.
[4] K. Bharat and A. Broder. *A technique for measuring the relative size and overlap of public web search engines*. In WWW7, 1998.
[5] A. Broder. *A taxonomy of web search*. In SIGIR Forum 36, 2002.
[6] C. Carpineto and G. Romano. *Concept Data Analysis: Theory and Applications. John Wiley & Sons, 2004*.
[7] H. Chen and S. T. Dumais. *Bringing order to the web: automatically categorizing search results*. In SIGCHI00.
[8] P. A. Chirita, D. Olmedilla, and W. Nejdl. PROS: *A personalized ranking platform for web search*. In Int. Conf. on Adaptive Hypermedia and Web-based Syst., 2004.
[9] B. Fung, K. Wang, and M. Ester. *Large hierarchical document clustering using frequent itemsets*. In SDM03.
[10] F. Giannotti, M. Nanni, and D. Pedreschi. *Webcat: Automatic categorization of web search results*. In SEBD03.
[11] J. Grabmeier and A. Rudolph. *Techniques of cluster algorithms in data mining. In Data Mining and Knowledge Discovery, volume 6(4), pages 303–360, 2002*.
[12] M. Halkidi, Y. Batistakis, and M. Vazirgiannis. *On clustering validation techniques. In JIIS, 2001*.
[13] T. Haveliwala. *Topic-sensitive pagerank*. In WWW12, 2002.
[14] M. A. Hearst and J. O. Pedersen. *Reexamining the cluster hypothesis: Scatter/gather on retrieval results.* In SIGIR-96.
[15] G. Jeh and J. Widom. *Scaling personalized Web search*. In WWW13, 2003.
[16] D. J. Lawrie. *Language Models for Hierarchical Summarization.* PhD thesis, Amherst, 2003.
[17] D. J. Lawrie and W. B. Croft. *Generating hiearchical summaries for web searches.* In SIGIR03.
[18] Y. S. Maarek, R. Fagin, I. Z. Ben-Shaul, and D. Pelleg. *Ephemeral document clustering for web applications*. Technical Report RJ 10186, IBM Research, 2000.
[19] M. Meila. *Comparing clusterings.*Technical Report 418, University of Washington, 2002.

