Abstract—Search engines are turn out to be the most helpful tools for gaining useful information from the Internet. One of the main defects of search engines are no matter how unlike the user is the search results referred to the same informational query keeps the same ignoring the difference of users interest. For this purpose we need personalization to obtain the results according to each users context. Thus there is a need of a highly efficient and effective ranking algorithm that provides search results according to user preferences. In this paper, we have proposed a new personalization algorithm. This new algorithm is a amalgamation of page rank algorithm and re-ranking algorithm. So it takes benet from both these algorithms. This approach also takes advantage of modified query expansion. Using term frequency of keyword according to users previous search history which is stored as a 3-level semi structured web log le and it is used for query expansion. Re ranking algorithm re-order the search result by comparing the result from Vector Space Model (VSM) and result from search engine. The re-ranking algorithm further classified the URLs according to dwell time, click history and last action performed. Finally it shows the effectiveness of proposed approach by using different experiments with different search strategies. The Result analysis shows that by using the re-ranking method based on user activities, the precision, recall and NDCG values for search results shows significant improvement.

Index Terms—Personalized Web Search, Information Retrieval, User Profiling, Re-Ranking Algorithm, Page Ranking.

I. INTRODUCTION

Now a days we can get everything from Internet. Internet is an inevitable thing in our life. World Wide Web is an information container. We can access data from WWW through search engines. Search engines produce information based on user query. Commonly used web search engines are Google, Yahoo, Amazon, Bing, Ask etc. Human being’s interests are different according to same things. Some of the queries are ambiguous in nature. According to a programmer the kitkat is an android version but in case of an ordinary user it is a chocolate. If person changes his interest also changes. So for case of an ambiguous query we need personalization.

The main goal of personalized search engines is to tailor search results to the needs of individual users based on their interests and preferences.

The search engines retrieve links match with user query. These links are ranked using some re-ranking algorithm. User interest supported result from search engine calculation starts from the query entering step first. We expand query using some techniques. In some cases add the synonymous and hyponyms of query words to this actual query and search based on this query gives more relevant result than previous simple query strategy.

We can get information from WWW using web crawlers. The queries entered in web search engines are used for information search. The results from search engines are ranked using ranking algorithms. There are a lot of ranking algorithms are proposed in different papers. In this paper, we discussed about different techniques used in personalization. Main classification is based on the ranking algorithm. The efficiency of ranking algorithm shows the efficiency of personalization.

Whenever user issues a query, in online phase, users query is first fed upto query expansion that recognizes and eliminate ambiguity in query and pass this expanded query to search engine. Search engine get backs result on expanded query and move into the proposed re-ranking system which re-rank the retrieved result by analyzing user interest model stored in form of VSM. VSM contains the interest value of particular user on particular links. User interest on particular link is calculated by how many times user had click this link, how much time user had spent in this link, actions performed on the link and actual rank of link given by the commercial search engine. If a link has Higher interest value it means that user has more interest in this link than other links.

This paper is organized as section 2 discuss about the different techniques for searching previously employed. Section 3 indicates proposed architecture for searching. Section 4 discuss about experiments conducted and its result obtained. This paper is concluded in section 5.

II. RELATED WORKS

Different approaches were proposed by researchers in the area of web search personalization. Some of these approaches are based on the users geographical location consideration the location factor only. In such approaches, the retrieved results are related to the users language and his demographic attributes, without considering any other user preferences. Although, these approaches may give better
results than the traditional search engines, the users in the same geographical area will have the same results even if they have different preferences and interests. On the other hand, other some approaches re-rank the retrieved search results from the traditional search engines based on the user preferences. The main disadvantage of these approaches is that the search process relies on the original search query without taking the user preferences into consideration. A user feedback can be an important factor to fine-tune the search results, thus another type of approaches employs the user implicit feedback to avoid direct user involvement. These approaches rely on the user feedback only, so the search process takes long time and passes through multiple iterations. Finally, the most famous and effective approaches build a robust user profile from different resources. This profile contains all user preferences, and hence it is used in web search personalization. The main disadvantages of these works are either they ignore the vocabulary problems or involve the user in enhancing and maintaining his profile. Ibrahim F. Moawad et al.,[1] proposed an agent based web search personalization using dynamic user profile. It mainly focus on building a user profile and updating it eventually. It uses Wordnet Ontology for Web search personalization. This technology can solve the vocabulary problems such as polysemy and synonymy. Here it uses a WSP conceptual model. This is proposed inorder to tackle the above problems. The detailed description is given in the following chapters.

Sun L et al.,[2] proposed an implicit user feedback approach to personalized search. This paper focus on implicitly collecting feedback from the user. The main advantage is that the use of implicit feedback saves time by exploiting user logs and also supports other languages other than English such as Chinese. But it is not so accurate as the explicit feedback and it gives only low performance. Micarelli et al.,[3] proposed a method for performing personalized search on World Wide Web, the adaptive web. It mainly focused on World Wide Web. The main advantage is that it increases the search engine accuracy and also reduces the search time in sorting. The drawback is that the Future behaviour of user is not specified and language semantics is not identified.

Stermsk et al.,[4] proposed a method for refinement of user profile using explicit user interest modelling. It relieves the user from time consuming and complex task to define his user profile. The main disadvantage is that there is no ontology. Another problem is that it can’t solve Vocabulary problems and refinement has to be done manually.

Leung K W et al.,[5] proposed a method for personalized web search with location preferences. It mainly focus on web searching based on geographical location. The main advantage is that it makes more sense and can find location related things faster. It has a disadvantage that if the search query is related to some other location it can’t give fruitful results.

III. PROPOSED APPROACH

In this section presents proposed approach that combines both the query expansion and re-ranking algorithm. Fig I shows our proposed approach for personalized web search. When user will search for any query, its first sent to query expansion. Query expansion attaches useful keywords from user profile and based on users search query and it also provide some other keywords that is utilized for general search for same. At time changes the interest of same user differences. Consequently different user have different context. Because of that the search history is not sufficient for recognize the current context of user. But in our proposed system, the keywords retrieved for general search is helpful to recognize the current interest of user on individual queries. Then, this expanded query is feed to search engine. The result from search engine is passed to Re-ranking algorithm, it will evaluate rank of each retrieved links and reorder them according to rank. The query expansion and re-ranking algorithm is described below.

A. Query Expansion

Co-occurrence based query expansion approach is used here. For example if we search an ambiguous query cricket, it considers words sport and insect related to this query. Therefore the word cricket has higher priority instead of other words in the context of co-occurrence approach. In this system arrange each word in a query into three levels according to their term frequency value. Term frequency is calculated by

$$tf_w = \sum_{k=1}^{n} \frac{|W_k|}{|W|}$$

(1)

Where $tf_w$ is term-frequency of keyword $w$, $|W_k|$ is no. of time keyword $W_k$ presents in set and $|W|$ is total occurrence of all the keyword in the set. Term-frequency of query cricket is denoted in TABLE I. Here Cricket is placed in level 1 according to its value. Term sport is placed in level 2.

For user profile creation, previous searching history of particular user is used. Previous history of user is stored in web log...
data. Each time user click on a particular link by his search, that link, its heading, query searched, actions performed on that link, snippets, how much time user spend on particular link etc are stored as a history of user. This is stored as a three level semi-structured format. From this title, snippet and URL is used for finding user interest list. Then, calculate term frequency of each keyword. If this keyword is present both in the user interest list and in the three level semi-structured formats created earlier and its frequency value is higher, then this keyword is denoted as a valuable keyword for further search.

Whenever a user will search a query, the system locates the level of particular query in the user interest list stored as a three level semi-structured format. After that it computed the weight of each lower level term. The weight is calculated by its number of Child. The highest number of child containing term is added with query word and this is forward to search engine for find personalized result (This result is displayed in the left side of search engine interface). The remaining terms are used for general search and it is displayed on the right of the search engine interface. The Fig II shows the result after search the query mouse. Personalized list shows the computer mouse and the general search list shows insect mouse.

B. Re-Ranking Algorithm

e-Ranking algorithm further arrange the result from searching by consider the rank of each link. Here Vector space model is used for document representation. Co-sign similarity function is used to get matched terms related to particular query from web log three level semi-structured data. Co-sign similarity of two vectors query and link terms is calculated using equation 2

\[
\cos \theta = \frac{d \cdot q}{||d|| \cdot ||q||}
\]  

(2)

After obtain the related terms about query from web log, bring out the link, the time spent on that link, action performed and number of times the particular user clicked that link. This link list along with above details and search result from search engine is send to re-ranking algorithm. Re-ranking algorithm first take the rank by page rank algorithm then calculate the Perform Rank(li) by Action Value(li), Click Value(li) and Dwell Time(li) of particular link. So the actual rank of the link is the sum of these two ranks. Then arrange the search result from search engine by this rank. This re-ranking algorithm is shown in Fig III.

Perform Rank(li) is the summation of Action Value(li), Click Value(li) and Dwell Time(li).

\[
\text{PerformRank}(l_i) = \text{Click value}(l_i) + \text{Dwell Time}(l_i) + \text{Action value}(l_i)
\]

(3)

where

- Click value(li) denotes the proportion of number of clicks made by user u on the page li for query 'q' with respect to total no.of clicks on all the pages made by user u for query ‘q’ given by eqn 4.
- Dwell Time(li) denotes the time spent by user ‘u’ on the page li given by eqn 5.
- Action value(li) denotes the action performed on the page li like saving, printing etc. given by eqn 6.

\[
\text{Click value}(l_i) = \frac{\text{No.ofclicksonpagelibyuseruforqueryq}}{T \cdot \text{otalno.ofclicksonallthepagesmadebyuseruforqueryq}}
\]

(4)

\[
\text{Dwell Time}(l_i) = \frac{\text{Timespentbyuseruonpageli}}{\text{Maximumtimespentbyuseruonanypageliinpast}}
\]

(5)

Further, There can be four types of actions that user can perform on any web page. The weight is assigned according to the relevancy of the action. As, if someone is printing the page means it has higher utility at present, Saving is less scored as the user will require it in future, bookmark come next in priority as the user don’t need the page currently. Sending comes last in priority list as page is used by some other user. Thus, the Action value(li) is sum of Printwt, Savewt, Bookmarkwt and Sendwt given in eqn 5.9.

\[
\text{Action value}(l_i) = \text{Print}(l_i) + \text{Save}(l_i) + \text{Bookmark}(l_i) + \text{Send}(l_i)
\]

(6)

Where

- Print(l) = log\{click(q, *, u) + 1\} \times n \times 4
- \( \text{Save}(l) = \log_2(\text{click}(q, *, u) + 1) \cdot n + 3 \)
- \( \text{Bookmark}(l) = \log_2(\text{click}(q, *, u) + 1) \cdot n \cdot 2 \)
- \( \text{Send}(l) = \log_2(\text{click}(q, *, u) + 1) \cdot n + 1 \)
- \( \text{click}(q, *, u) \) is the no. of pages viewed for the query \( q \) by user \( u \).
- \( n \) is the no. of times action performed.

**Re-Ranking Algorithm**

Input: \( Q \)
Output: \( O' \)

\( O = \{o_1, o_2, o_3, ..., o_m\} \) Result from Search Engine

\( O' = \{o_1, o_2, o_3, ..., o_n\} \) Re-ranked Result

Re-ranking module:

\[
\begin{align*}
\text{For each document in } O, \text{ do} & \\
\text{Rank} &= \text{Page rank} () \\
\text{Perform Rank} &= \text{Click Value} () + \text{Dwell Time} () + \text{Action Value} () \\
\text{Actual Rank} &= \text{Rank} + \text{Perform Rank} \\
\end{align*}
\]

Arrange each document in \( O' \) by rank;

Return \( O' \) to the search engine interface;

**TABLE II** shows the list of links and their ranks retrieved by search engine when User issued a query “Apple”. It first processed by query expansion, append some keywords as discussed in section 3.1. So the result retrieved in TABLE II is of “Apple Fruit” where the term “Fruit” is appended by query expansion technique.

**TABLE IV**

<table>
<thead>
<tr>
<th>Link</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="http://www.besthealthmag.ca">http://www.besthealthmag.ca</a></td>
<td>1</td>
</tr>
<tr>
<td><a href="http://en.wikipedia.org">http://en.wikipedia.org</a></td>
<td>2</td>
</tr>
<tr>
<td><a href="http://www.nutrition-and-you.com">http://www.nutrition-and-you.com</a></td>
<td>3</td>
</tr>
<tr>
<td><a href="http://urbanext.illinois.edu">http://urbanext.illinois.edu</a></td>
<td>4</td>
</tr>
<tr>
<td><a href="http://www.huffingtonpost.com">http://www.huffingtonpost.com</a></td>
<td>5</td>
</tr>
<tr>
<td><a href="http://www.huffingtonpost.com">http://www.huffingtonpost.com</a></td>
<td>6</td>
</tr>
<tr>
<td><a href="http://www.vermontapples.org">http://www.vermontapples.org</a></td>
<td>7</td>
</tr>
<tr>
<td><a href="http://http://www.britannica.com">http://http://www.britannica.com</a></td>
<td>8</td>
</tr>
<tr>
<td><a href="http://nutritiondata.self.com">http://nutritiondata.self.com</a></td>
<td>10</td>
</tr>
</tbody>
</table>

**TABLE V**

<table>
<thead>
<tr>
<th>Link</th>
<th>Rank</th>
</tr>
</thead>
<tbody>
<tr>
<td><a href="http://http://www.whfoods.com">http://http://www.whfoods.com</a></td>
<td>1</td>
</tr>
<tr>
<td><a href="http://www.nutritiondata.self.com">http://www.nutritiondata.self.com</a></td>
<td>3</td>
</tr>
<tr>
<td><a href="http://http://www.vermontapples.org">http://http://www.vermontapples.org</a></td>
<td>4</td>
</tr>
<tr>
<td><a href="http://http://www.besthealthmag.ca">http://http://www.besthealthmag.ca</a></td>
<td>6</td>
</tr>
<tr>
<td><a href="http://en.wikipedia.org">http://en.wikipedia.org</a></td>
<td>7</td>
</tr>
<tr>
<td><a href="http://urbanext.illinois.edu">http://urbanext.illinois.edu</a></td>
<td>8</td>
</tr>
<tr>
<td><a href="http://www.nutritionandyou.com">http://www.nutritionandyou.com</a></td>
<td>9</td>
</tr>
<tr>
<td><a href="http://nutritiondata.self.com">http://nutritiondata.self.com</a></td>
<td>10</td>
</tr>
</tbody>
</table>

In TABLE III, we have shown links that user had previously searched, Dwell time indicates how much time the user has spent on individual links. In TABLE IV, we have shown links that user had previously searched, and action value based on previous actions. From TABLE IV it has been observed that the link “http://www.besthealthmag.ca” has spent more time than other ones. This links along with links in TABLE II is input for our proposed re-ranking algorithm.

Finally, based on interest value the results are re-ordered and sent to user. TABLE V shows the result after applied re-ranking algorithm. So it has been observed, if user has spent more time and action performed it has been displayed top of the results. It has also been observed that “http://www.nutrition-and-you.com” has higher Dwell time than “http://en.wikipedia.org”, but “http://www.nutrition-and-you.com” has low action value though Wikipedia is displayed at top of the nutrition because we have also considered the actual rank of link and in this case rank of Wikipedia is higher than nutrition as shown in TABLE II.

**IV. EVALUATION**

Dataset is collected from user’s click through data. We have accepted different quires from users with their expectation on query explicitly. Based on expectation first user will search for query, for instance if some user want to search for “apple fruit” then he will make search query regarding the apple fruit. These all information would be stored in web log data. After that we have asked the user to enter some ambiguous query regarding their previous search goal. Finally we have shown that our proposed system has given better search result that match with their expectation and we have calculated the precision@n i.e. the fraction of top n document that are relevant. In our analysis n is 30 i.e. fraction of top 30 document that are relevant.
A. Tools Used

Different tools used are Java Server Pages, Eclipse IDE, MySQL.

B. Measures

The following measures are used as an evaluation criteria for analyse the effectiveness of the system:

1) Top-n Precision: It is the precision for ‘R’ relevant documents. This ‘R’ is employed for cutoff for calculation. Precision (p) is the fraction of the documents regained that are relevant to the user’s information need.

\[
\text{Precision} = \frac{\text{number of interesting pages retrieved}}{\text{total number of pages retrieved}}
\]

Higher the precision implies better quality of algorithm in retrieving relevant pages where as recall gives quantitative analysis. An algorithm is better if both the precision and recall values are high.

2) NDCG Analysis: We have also evaluate NDCG@n for our result analysis. NDCG is normalized Distributed cumulative gain is used to calculate the relevancy of retrieved document by considering rank of documents. Our proposed re-ranking algorithm re-ranks the search results by calculating interest value of individual link. Higher the interest value of link is more relevant for user and should be displayed at top to the search results. Precision only calculate the relevance of documents. Now for instance if some user U1 get all the relevant results but the link user had searched frequently is displayed at 20th position then precision will not be change. So most relevant result at top of list or at bottom of the list precision will not be changed so we will evaluate effectiveness of our proposed algorithm by calculating NDCG@20 for same results.

\[
\text{NDCG} = \frac{\text{DC}_{tt}}{\text{IDC}_{tt}}
\]

3) False Discovery Rate: The False discovery rate (FDR) is one way of conceptualizing the rate of type I errors in null hypothesis testing when conducting multiple comparisons. FDR-controlling procedures are designed to control the expected proportion of rejected null hypotheses that were incorrect rejections

\[
\text{False Discovery Rate} = \frac{\text{number of irrelevant pages retrieved}}{\text{total number of pages retrieved}}
\]

C. Performance Analysis

TABLE VI shows the precision@30 of general search and personalized search for different queries with different expectation and TABLE VII shows the recall of general search and personalized search for different queries with different expectation. There are 6 user enters the different or same query with different expectation. As shown in Fig. IV relevancy of search engine is very low when first time search We have assumed that all that user had previously searched for the query for their context. So if user have same query like apple with different context i.e. fruit and computer, than in commercial search engine will give more relevant results to user whose context is computer rather than fruit. Whereas in proposed system will able to give more personalized results in any of the context of users.

<table>
<thead>
<tr>
<th>User</th>
<th>Query</th>
<th>Context</th>
<th>General Search</th>
<th>Personalized Search</th>
</tr>
</thead>
<tbody>
<tr>
<td>U1</td>
<td>Apple</td>
<td>Fruit</td>
<td>0.133</td>
<td>0.83</td>
</tr>
<tr>
<td>U2</td>
<td>Opera</td>
<td>Music</td>
<td>0.066</td>
<td>0.76</td>
</tr>
<tr>
<td>U3</td>
<td>jaguar</td>
<td>Cat</td>
<td>0.033</td>
<td>0.845</td>
</tr>
<tr>
<td>U4</td>
<td>Apple</td>
<td>Computer</td>
<td>0.6</td>
<td>0.865</td>
</tr>
<tr>
<td>U5</td>
<td>Opera</td>
<td>Browser</td>
<td>0.866</td>
<td>0.8965</td>
</tr>
<tr>
<td>U6</td>
<td>jaguar</td>
<td>Car</td>
<td>0.733</td>
<td>0.8865</td>
</tr>
</tbody>
</table>

**TABLE VI: PRECISION @30 FOR DIFFERENT QUERIES WITH DIFFERENT EXPECTATION**

<table>
<thead>
<tr>
<th>User</th>
<th>Query</th>
<th>Context</th>
<th>General Search</th>
<th>Personalized Search</th>
</tr>
</thead>
<tbody>
<tr>
<td>U1</td>
<td>Apple</td>
<td>Computer</td>
<td>0.133</td>
<td>0.83</td>
</tr>
<tr>
<td>U2</td>
<td>Opera</td>
<td>Music</td>
<td>0.066</td>
<td>0.76</td>
</tr>
<tr>
<td>U3</td>
<td>jaguar</td>
<td>Cat</td>
<td>0.033</td>
<td>0.845</td>
</tr>
<tr>
<td>U4</td>
<td>Apple</td>
<td>Computer</td>
<td>0.6</td>
<td>0.865</td>
</tr>
<tr>
<td>U5</td>
<td>Opera</td>
<td>Browser</td>
<td>0.866</td>
<td>0.8965</td>
</tr>
<tr>
<td>U6</td>
<td>jaguar</td>
<td>Car</td>
<td>0.733</td>
<td>0.8865</td>
</tr>
</tbody>
</table>

**TABLE VII: RECALL FOR DIFFERENT QUERIES WITH DIFFERENT EXPECTATION**

Then evaluate NDCG@n for result analysis. NDCG is normalized Distributed cumulative gain is used to calculate the relevancy of retrieved document by considering rank of documents. Proposed re-ranking algorithm re-ranks the search results by calculating interest value of individual link. Higher the interest value of link is more relevant for user and should be displayed at top to the search results. Precision only calculate the relevance of documents. Now for instance if some user U1 get all the relevant results but the link user
had searched frequently is displayed at 20th position then precision will not be changed. So most relevant result at top of list or at bottom of the list precision will not be changed so we will evaluate effectiveness of proposed algorithm by calculating NDCG@20 for same results.

TABLE VIII shows the links that user had previously searched. Links has given three score based on their relevancy. 3 indicate more relevant, 2 indicate relevant and 1 indicate less relevant. TABLE IX shows the NDCG@20 for queries Apple, Opera and Jaguar with the context fruit, music and cat. We have assumed, if user will spend more time and perform actions based on priority on particular link than it is more relevant and given higher score to that link. It has been observed from the Fig.VI, NDCG@20 is small if user had clicks on limited links but proposed approach will better reorder the result in context of user. As user will click on more no. of links NDCG is incremented gradually. Fig. VII shown NDCG@20 with respect to no of relevant links clicked by the user. Fig. VII is also shows that if user have increased their relevant links NDCG@20 is incremented and at some saturation point (in this case when R=20), the NDCG@20 becomes 1 for that particular user. Proposed system displayed the more relevant result at top of the retrieved list to better give personalized search result to user. This can be proved by the Fig. VIII that shows the cumulative gain (CG) of top ten result of query apple. CG can be calculated by the equation

\[ C_{t_1} = \frac{2^n - 1}{\log_2 i + 1} \]  

Where n indicate relevant score of the link at position i. We have already specified the score in TABLE VII. As we can see that relevance score for both the links in the table is same though it has been ordered at different position. This can be achieved by considering the actual rank of the links like: For link at 1st position

\[ C_{t_1} = 23 \frac{1}{\log_2 1 + 1} = 7 \]

For the link at 2nd position

\[ C_{t_2} = 23 \frac{1}{\log_2 2 + 1} = 4.441 \]

In proposed system considered the actual rank of the links that is useful to calculate the exact interest value of the user and also avoid the conflict the reorder the two links which has higher relevance score. So from Fig. VIII link1 and link 2 has higher relevance score but the rank is always different that reorders the retrieved set of results.
The conclusion of this project was to develop a system that provides better personalized result. This system introduced an approach that identifies and removes the ambiguities from user’s query by appending some useful keywords. The algorithm that re-order the user’s search result based on dwell time, click value and action performed can personalize the result from search engine. This system also removes irrelevant document and their by decrease the time for searching with unwanted result. The interest of user changes according to time, if one has interest in searching opera browser, after sometime his preference will be changed to opera music. Then the right side of the interface also shows the general search result that can catch the current interest of user efficiently. 

The result analysis shows that the proposed system personalized the search result according to particular user’s interest and also shows that it has above 50% improvement in personalized result compared to general search. The proposed system also shows relevant result at top of the result. The search engines use keyword matching rather than semantic matching i.e. synonym of a keyword is not taken into consideration while searching for result and lack of using personal information such as hobbies, interest and preferences of user. So as a future work I propose query expansion based on the synonyms of particular keyword and also take feedback from user (insert a like icon). This feedback value is added to calculate rank by re-ranking algorithm. This feature also shows that the most relevant document according to all user. The most liked page considering all web users is also suggested to all users for their usage.

It must be noted that this system can’t provide personalized result in the first search of individual user. For that purpose, we must provide a searching strategy based on the interested area provided by the user in the login section for the first time. Weighted page ranking algorithm is used instead of calculating absolute rank using page ranking algorithm. It is because page ranking algorithm only consider inbound and outbound links and page rank distributed equally among all linked pages. But in weighted page ranking algorithm the page rank is distributed based on their importance also.

REFERENCES


