A FRAMEWORK FOR PERSONALIZATION USING QUERY LOG AND CLICKTHROUGH DATA

Vijayalakshmi. K¹, Dr. Sudarson Jena², Dr. D. Vasumathi³, Dr. R. Rajeswara Rao⁴

¹Dept of CSE, JNTU, Hyderabad
²Dept of IT, Gitam University, Hyderabad
³Dept of CSE, JNTU H, Hyderabad
⁴Dept of CSE, JNTU, Vijayanagaram

ABSTRACT

Personalization of a web page involves dynamically altering the contents of the web pages according to the preferences or interests of users for retrieving, appropriate information for a given query. Although many personalization algorithms have been explored, but it is still not certain whether personalization is really helpful all time on dissimilar queries for different users and under diverse searching scenarios. Most of the personalization approaches are based on the user query logs or user profiles. Personalized web search only using query logs may not be effective and not be according to a user's preferences. This paper proposes a novel framework for search result personalization based on user query log and clickthrough data. The proposed framework implements a re-ranking approach and generates personalized result with high relevancy in information retrieval. The re-ranking approach combines users search context and users browsing manners resulting in effective information retrieval. This framework derives an extended set of conceptual and relevance preferences of a user based on the extracted log and clickthrough data concepts for the search. Experiment evaluation results show that the framework and re-ranking approach is highly effective for result personalization on web search and information retrieval.

Keywords: Search Engine, Personalization, Clickthrough Data, Information Extraction, Query Log.

1. INTRODUCTION

The Internet has changed the way humans interact with information. The excess amount of information has made a difficult and time consuming task the human processing and filtering through the available information to find what human users are looking for. When users search for information, they would like to receive the most likely results to satisfy their query first. But,
generally user describes their needs of information in very few keywords or in short phrases, which creates challenging in information retrieval in a web context due to the vast and scalable contents on the web. It stirs large scale querying and browsing activity that is of more interest to users over a sizable window period, which is unusual relative to normal patterns of querying and browsing behavior. Many techniques and approaches [1][2][3] are discovered and evaluated in different patterns from user web usage logs for improvising web personalization.

Web usage mining analyzes the behavior of users according to the data recorded in search engine and Web site access logs. A large amount of search logs accumulated by web search engines in form of user clickthrough data. These logs typically contain user-submitted search queries, followed by the URL of Web pages, which are clicked by users in the corresponding search result page. Although these clicks do not reflect the exact relevance, they provide valuable indications to the users’ intention by associating a set of query terms with a set of web pages. If a user clicks on a web page link, it is likely that the web page link is appropriate to the query, or at least related to some extent. Several applications have been proposed along this direction, such as term suggestion [9], query expansion [7], and clustering of query [5][6].

Current information extraction systems (or search engines) returns big list of matching results based on keyword. However, users typically view only top few result documents from the big list of results extracted by the search engines [8][15]. This demands to present the most appropriate result information documents on top to get better user satisfaction. However, without user information need knowledge this task is difficult, because “relevance” of a document depends on his/her user and the individual query. A dilemma is observed for learning with preference, for example, query \(Q_1\) as “Rock” of user \(U_1\), document \(D_1\) has higher preference than document \(D_2\) and, for the same query of user \(U_2\), document \(D_2\) has higher preference than document \(D_1\). In such scenario ranking of \(D_1\) and \(D_2\) documents creates a dilemma. So, the major challenges for personalized search are modeling appropriate user context and learn a user model to improve search accuracy.

Even though the idea of personalization is not new, but in this paper we contribute two aspects. Firstly a novel framework model utilizing user query logs and clickthrough data for search result personalization and secondly an efficient re-ranking approach to extract relevance results in relevance to the user query.

Our approach finds most appropriate result documents for a user based on a given query. Here, our focus is to evaluate the effective association between User Queries and Clickthrough data and customizes search results according to each individual preferences/interests. The re-ranking procedure finds the score of the web pages that are retrieved from different search engines.

The paper organized in following sections as, Section 2 presents the related work on personalized web search. In section 3, we present the proposed framework and re-ranking procedure for personalized information retrieval that combines users’ context and browsing behavior. Section 4 describes the experimental results and section 5 concludes the paper.

2. RELATED WORK

Personalized web search need to modify the information and the organization of a website or search result to present the most relevant web resources to the specific and individual needs. It can be achieve by following the user navigational behavior, as it can provide useful information of user interest and individuality. Several attempts are made [4][9][10] for personalizing Web search which focuses on personalized search strategies to meet user interest based on their past activities.

2.1 Query Logs

Query logs are auto saved data of user activities on search engines servers. It consists of user identity attributes as Session ID, IP address, Time-stamp, Query string, Number of results on results
page and Results page number. A relevance clickthrough data also saved consisting of clicked URL, associated query, position on results page and Time-stamp attributes in the log. The application used in client side can be modified to handle the query and clickthrough usage logs in the user side computer. It can be very important source for user personalization. There has been some work related are described below.

K. Sugiyama et al. [14] investigates on the web navigated history data in previous few days for personalization search. Based on the time they segregated the navigated history data into three partitions according to the clickthrough time stamp, as before today, today data but these data are not includes current session data. It was observed that the relevance performance of using web navigation history is more effective in compare to utilizing the user feedback. M. Speretta et al. [13] also utilized the users search history behaviour to create user personalization profiles of interest. A number of other works are investigated in [4][10][11][12] to make use of past navigated log data and queries for mining clickthrough logs data to construct efficient user personalization information.

2.2 Language Modeling Approaches

Language modeling based approach for information retrieval has developed within the past years as a novel probabilistic framework for investigating information retrieval processes [1]. This approach represents a structural representation that describes the main topics and their organization within a given domain of discourse. Modeling language structure is particularly relevant for domains that exhibit recurrent patterns in content organization, such as news and encyclopedia articles. Computing language models for an arbitrary domain is a challenging task due to the lack of explicit, unambiguous structural markers. Moreover, texts within the same domain may exhibit some variability in topic selection and ordering this variability further complicates the discovery of language structure.

Applied to information retrieval and language modeling describes the difficulties of estimation the probability of a query and a information document might have been generate or more simply, content modeling describes to the activities of estimating a possibility of distribution that capture the statistical standard of the contents used. There has been a lot of work [12] are related to language modeling for information retrieval and related applications. There has not been much work in to auto capturing the information about the user and environment of the retrieval process in spite of the many progress made in language modeling and information retrieval context.

Shen et. al [21] proposed a decision theoretic framework for implicit user modeling for personalized search. They consider the short term context in modeling user. A language model is computed from the short term history and is used to improve the retrieval performance. In [15], long term search history of the users is mined and language models are computed which are then used to improve retrieval performance.

2.3 Collaborative Filtering Based Approaches

The growing popularity of social network demands an improved web search to group the similar interest users. A huge amount of history data are logged in history of community sites on daily users interactions. An exploitation of query repetition among the users within the community is needed. Similar queries and their corresponding clicked documents are used to either recommend documents or are used in improving search results. Different approaches have been proposed varying the way communities are defined and employing different learning techniques for discovering communities, user and community profiles.

Chidlovski et. al [18] describes the architecture of a system performing collaborative re-ranking of search results. The user and community profiles are built from the documents marked as relevant by the user or community respectively. These profiles essentially contain the terms and their appropriate weights. Re-ranking of the search results is done using the term weights using adapted
cosine function. The searching approach and the result ranking of the relevant information documents are achieved within the circumstance of a particular user or community interest.

Armin Hust [19] performed query expansion by using previous search queries by one or more users and their relevant documents. This query expansion method reconstructs the query as a linear combination of existing old queries. The terms of the relevant documents of these existing old queries are used for query expansion. However, the approach does not take the user into account.

Lin et. al [20] presented an approach to perform personalized web search based on Probabilistic Latent Semantic Analysis (PLSA), a technique which stems from linear algebra. They extracted a co-occurrence triple consisting of a user, query, and corresponding web pages viewed for a query, by mining the web-logs of the users and modeled the latent semantic relationship between them using PLSA.

PLSA technique is based on Expectation-Maximization (EM) algorithm. It measures the object association relationship between the hidden factors and two sets of objects using probability estimation value. PLSA has been successfully used in different type of application domain such as text learning and information retrieval. It also successfully used in web usage mining. PLSA based approach framework discovers and analyzes the web navigation patterns for personalized search. In this paper we compare our approach with PLSA-Based approach [20] for experimental evaluation.

2.4 Machine Learning Based Approaches

Personalized search is one such potential problem where the use of Machine learning algorithms have received a wide attention recently to learn functions that can perform desired operations when trained on required amount of data. The previous history of the user, can we learn a model representing the user. This makes user modeling a perfect application for machine learning.

Joachims et al. [16] has proposed a new machine learning algorithm Ranking SVM, a variation of the Support Vector Machine learning algorithm which can learn from the partial feedback data present in the clickthrough data of users. Teevan et.al [17] proposed a personalized search using user profile based on desktop search index and to learn the user profile it use ranking Nets. The evolution of the proposal considers user profile as implicit user feedback and integrates them for the ranking search results.

Aslam et al. [22] proposed Bayes-fuse method based on Bayesian inference. It works on machine learning approach using training data for result merging. It considers the local rank of the results returned by the search engines as the evidence of relevance for result merger. The result relevancy and irrelevancy are given based the probability of relevancy \( P_{rel} \) and irrelevancy \( P_{irr} \) computed against the number of search engine selected for the query and based on the optimal retrieval principle in information retrieval the result with highest ratio \( O_{rel} = \frac{P_{rel}}{P_{irr}} \) are given as a highest relevant. As a machine learning approach for personalized search ranking with original web ranking can achieve better results than the original ranking as observed in previous related works. In this paper we compare Bayes-fuse method also with our approach to evaluate the effectiveness of the proposal.

3. QUERY PROCESSING AND PERSONALIZATION

In this section, we describe our proposed key components of the framework and re-ranking approach for personalization.

3.1 Proposed Framework

Our framework (Fig. 1) implements a re-ranking process and enables an effective personalization using user query log and clickthrough data. The framework consists of five components: Request Handler, Query Processor, Result Handler, Event Handler and Response
Handler. It has log repositories which stores user query logs and clickthrough data, the format of data structure is explained in section 4. In figure 1, dashed arrow represents the data transaction with database and solid arrow represents the process communication between the components. Square block represents as components and circle as internal process blocks of a components in the framework.

**Result Handler** is the core component of this framework. It receives the search engine result lists and implements the re-ranking approach (explained in section 3.1) using query log and clickthrough data and generates personalized result which sends to the user. The **Request Handler** component handles the user query request and maintain request load by queuing the request. **Query processor** is a key component of the framework which process the user query and prepare keyword phrase which pose to search engine. It also updates the userid, query and keywords in the Query Log database. **Response Handler** presents the personalized results received from result handler. **Event Handler** is an event listener which listen the click events on results link. It captures the link URL and rank from the clicked page and update the clickthrough database.

![Fig –1: Framework for Personalization Using Query Log and Clickthrough Data](image)

When a query submitted by a user it received by the request handler. The submitted query might not be in appropriate structure for submitting to a search engine. Query Handler process this query to filter and prepare the keywords and phrases which are submitted to the search engine, and at the same time it update the Query log database. On retrieval of search result from search engines Result Handler filter the duplicate result and organize the result as per the search engine ranking. The organize result under goes re-ranking process with support of Query log and clickthrough data recorded for this user query. Execution of re-ranking process reorder the organized data as per the user passed interest and an appropriate personalized result presented to the user. The presenting of personalized result handles by result handler. It may possible the presented result may not be so relevant to user needs. To make it more precise each result link bounds with a click event. On clicking a particular link on the page event handler listen and records the clicked link data to
clickthrough database. The continuous of these activities improve the personalization relevancy as clickthrough data against a query increases for a user.

The proposed framework utilizes clickthrough data that is saved in search engine logs to suggest user behaviour and interest in web information searching. In general, when a user pose a query, the user usually navigates the entire result links list from top to bottom in a page. User generally clicks one or more result link that looks appropriate and relevance and skips those links which are not relevant. Effective information retrieval is achieved when a precise personalization approach perform re-ranking of the relevant links and place it in higher in results list. Therefore, we utilize user clicks as relevance decision measure to evaluate the searching accuracy. Since clickthrough data can collect straightforward with less effort, it is possible to do required behaviour and interest evaluation implementing this framework. Moreover, clickthrough data shows the actual real world distribution of user search interest queries, and searching scenarios. Therefore, using clickthrough data makes a closer real time personalization requirement cases in compare user feedback survey.

3.2 Proposed Re-Ranking Approach

The presented framework implements the re-ranking algorithm based on an assumption that user submit a query $Q$ for an information, and the obtained web results links are very frequently clicked by user ($U$) in the earlier sessions are more appropriate to user ($U$), than those links rarely clicked by $U$. Thus, a new ranking value of the resulted link $L$ which is retrieved on posing the query $Q$ by user can be computed.

Let’s assume that a result set of $S_R$ has obtained top 10 results on posing a query $Q$ as $(R_1, R_2, R_3, R_4, R_5, R_6, R_7, R_8, R_9, R_{10})$ and same as the current rank from $R_1 \ldots R_{10}$ as return by search engine.

An assumption, that a high numbers of clickthrough log data is recorded with respect to users in log database. An extraction of relevant clickthrough records as $C_S$ will be made based on the user identification as $U$ and query $Q$ as shown below in equation (1).

$$C_S = \exists(U) \cap \exists(Q)$$  \hspace{1cm} (1)

The search result, $S_R$ obtains from search engine an average rate of as $R_{rate}$ need to calculate using $C_S$ for re-ranking the obtained result. To compute the $R_{rate}$ an iteration of $S_R$ result need to made against $C_S$ data records, as shown below in equation (2),(3) and described in Fig-2.

$$R_r = \sum_{i=1}^{n} R_i$$  \hspace{1cm} (2)

$$R_{rate} = \frac{R_r}{n} \times 100$$ \hspace{1cm} (3)

Where, $R_r$ is the sum of rate count for record $R_i$ and $n$ is the number of records in $C_S$.

Based on the $R_{rate}$ of the search results the result will be sorted in descending order to user personalized result as $P_{res}$.

$$P_{res} = Sorting(S_{R_{1-n}}, R_{rate_{1-n}})$$ \hspace{1cm} (4)

Where, $S_{R_{1-n}}$ is the search record from $i=1$ to $n$ as number of search records in $S_R$ and $R_{rate_{1-n}}$ is the average rate of the each record from $i=1$ to $n$. 

122
4. EXPERIMENT AND RESULTS

To evaluate the search performance of personalized framework, each participating user is necessary to issue a definite number of test queries, and also need to decide whether each result is appropriate or not. An improvement of this mechanism make user to judge the most appropriate and relevant documents for the presented personalized results. However, limitation on the number of users and queries for test may give uncertainty for the result evaluation on accuracy and relevance of the personalization algorithm. As mention above we compare our approach with PLSA-Based and Bayes-fused based approach to measure the effectiveness.

4.1 Clickthrough Data

For obtaining clickthrough data, we have developed a web application with click event handler to store the clickthrough event link data using five different users’ clickthrough activities. We have taken 5 different queries as, “Web Personalization, Deep Web data Extraction, Data Mining for Information Service, Web Information Retrieval and Extracting Quality Data” and integrated with Yahoo search engine for evaluation.

As the framework described in Fig-1, each query processed and passed to the search engine and the resulted links are presented for browsing. User clicks the appropriate links which are relevant to the query, on click the relevant link the embedded click handler in the page handles the event and stores the link information in the database. Clickthrough data are denoted as a triplet \((q, r, c)\), where \(q\) is the input query consisting of a set of keywords, \(r\) is a list of ranked links, \((l_1, \ldots, l_n)\), and \(c\) is the set of links that the user has clicked on. Along with the triplet it stores the user system ID and result rank. We repeat this process to build a clickthrough database up to 10000 records. The records are stored in the database in columns as shown in the Table 1.
4.2 Evaluation and Measures

We first evaluate our proposed framework offline before evaluating other approaches. Then, we evaluate our approach with other approaches and compare with different measuring metrics as Precision, Recall and Fallout Rate.

**Precision Percentage Rate (P):** Precision in the information retrieval is used to measure the preciseness of a retrieval system. Precision is calculated as the ratio of Number of appropriate and relevant results against the Number of appropriate results. Hence it represent as,

\[
Precision (P)\% = \frac{\text{Number of appropriate and relevant results}}{\text{No. of appropriate Result}} \times 100
\]

**Recall Percentage Rate (R):** Recall measures the effectiveness of a query system in the information retrieval. It is calculated as the ratio of Number of appropriate and relevant results against the Number of relevant results, which represented as,

\[
Recall (R)\% = \frac{\text{Number of appropriate and relevant results}}{\text{No. of Relevant Result}} \times 100
\]

**Fallout Percentage Rate (F):** It measures error rate in the information retrieval. It is calculated as the ratio of Number of appropriate and nonrelevant results against the Number of nonrelevant results. It represented as,

\[
Fallout (F)\% = \frac{\text{Number of appropriate and nonrelevant results}}{\text{No. of Nonrelevant Result}} \times 100
\]

4.2.1 Framework Evaluation with Clickthrough Data

Firstly, we evaluate the proposed framework with a query as “Web Personalization” to a search engine, the return results are handled by result hander and implements re-ranking process for personalization. We collect top 20 results as shown in Table-2 and their rank order and perform a rule-based classification query on clickthrough database to obtain the relevance results. The rule implements a precondition and a matching attributes to classify the required results based on input as SystemID and User Query. We define a rule to obtain the clickthrough data as \( C_R \) as follows,

\[
C_R = (\text{SystemId='input systemid'}) \land (\text{Query = 'input User Query'})
\]

If the condition in the rule is true for the given attributes then only it returns the stored results. The obtained result of \( S_R \) and \( C_R \) are used for re-ranking process. Table-2 shows the search results obtained for the query “Web Personalization” using a search engine and a personalized result which is processed with the proposed re-ranking approach and clickthrough data. It shows an improved appropriate search results in relevance to the user needs based on it past activities.
Table 2: Query Search Results and processed Personalized Results

<table>
<thead>
<tr>
<th>Search Engine Result</th>
<th>Personalized Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>11. <a href="http://www.en.cyclopaedia.net/wiki/Web-personalization">www.en.cyclopaedia.net/wiki/Web-personalization</a></td>
<td></td>
</tr>
<tr>
<td>13. <a href="http://www.academia.edu/Documents/in/Web_Personalization">www.academia.edu/Documents/in/Web_Personalization</a></td>
<td></td>
</tr>
<tr>
<td>14. <a href="http://www.slideshare.net/web-mining-for-web-personalization">www.slideshare.net/web-mining-for-web-personalization</a></td>
<td></td>
</tr>
<tr>
<td>15. <a href="http://www.facebook.com/pages/Web-Personalization">www.facebook.com/pages/Web-Personalization</a></td>
<td></td>
</tr>
<tr>
<td>17. <a href="http://www.boxesandarrows.com/personalization/web-personalization">www.boxesandarrows.com/personalization/web-personalization</a></td>
<td></td>
</tr>
<tr>
<td>19. <a href="http://www.chasetheband.com/web-personalization">www.chasetheband.com/web-personalization</a></td>
<td></td>
</tr>
<tr>
<td>20. <a href="http://www.panovistamarketing.com/web_personalization.html">www.panovistamarketing.com/web_personalization.html</a></td>
<td></td>
</tr>
</tbody>
</table>

4.2.2 Performance Measures of the Framework

![Precision Performance](image1)

**Fig-3:** Framework Precision Performance with clickthrough data at different scale of results

![Recall Performance](image2)

**Fig-4:** Framework Recall Performance with clickthrough data at different scale of results
Fig-3 and 4 shows the precision and recall performance at different scale of search results in support to clickthrough data. The result shows an improvisation in precision rate with increasing number of clickthrough and rate of fall in recall rate. It proves the motivation of the proposal to provide better appropriate results to meet the user satisfactory level. In different scale of search results it performs very effectively as clickthrough approach is based on the previous clicks, the high number of repetition ratios in real query logs improvises the personalization performance.

We continue our performance evaluation with comparison with PLSA-Based and Bayes-fused based approach to support the framework performance. Here we refer a database of 10000 clicked records for our approach.

4.2.3 Performance Comparison with PLSA and Bayes-fused Approach

We run the experiment of different approaches online using Yahoo search engine to collect the results. In order to support our approach we build a database of 10000 clickthrough records for evaluation. With varying the result collection from 40 to 280 results we measure the precision, recall and fallout performance comparisons as shown in Fig-5, 6 and 7.

**Fig-5: Precision Performance Comparison**

**Fig-6: Recall Performance Comparison**
**Precision Rate:** Our approach shows an improvement in precision measure over PLSA and Bayes-fuse approaches. It shows an average improvement of 22% on 40 and 11% on 280 results against PLSA and 26% on 40 and 22% on 280 results against Bayes-fuse approaches. This is because of the support of clickthrough data reference and re-ranking approach which filter out non-relevant results for making more precise result in personalization.

**Recall Rate:** As recall rate measure the effectiveness of a query system in the information retrieval. Our approach shows a low recall rate measure over PLSA and Bayes-fuse approaches. It shows an average low of 26% on 40 and 11% on 280 results against PLSA and 37% on 40 and 22% on 280 results against Bayes-fuse approaches. This is because precision and recall are inversely proportionate and as precision show high on increasing results in recall rate shows low. It shows some increment with high number of results, it is due to fall of precision level.

**Fallout Rate:** Fallout rate measures error rate in the information retrieval. The result comparison of fallout rate shows an increment, it is because of high number of irrelevant results retrieved. In compare to the result our approach shows an low fallout rate compare to PLSA and Bayes-fuse. It is observed that our approach shows a 27% on 40 and 10% on 280 results low against PLSA-Based and 38% on 40 and 25% on 280 results low against Bayes-fuse approach.

As a conclusion we conclude that the proposed framework shows an improvisation on precision with clickthrough data and low recall and fallout rate for better personalization.

### 6. CONCLUSIONS

A search engine allows searching the entire web for information in its scope. But the huge increase of information over web makes challenging to search engine also to relate the relevant and appropriate results against the query. Results which are very appropriate and relevant may be ranked low for a specific query in the search engine search result list due to lack of sufficient parameters required to relate the relevancy and re-ranking the results in relevancy to the query. In this paper, we proposed an evaluation framework and novel re-ranking approach based on query logs and clickthrough data to enable support and evaluation of personalization search. To evaluate the framework and approach we create a clickthrough database on various different queries and run an evaluation test query for re-ranking and personalization result and also we compare the performance with PLSA-Based and Bayes-fused approaches. The experiments results shows that the proposed...
framework and re-ranking approach with click-based personalization worked very efficient in satisfying the user needs and also minimize the navigation searching time. The proposed algorithms efficiently work for re-ranking and user personalization. Our future work focuses on online evaluation of the personalized framework and development of an automatic prediction algorithm based on clustering the user profile, clickthrough data and user feedback.

REFERENCES


[6]. Zhicheng Dou, Ruihua Song, Ji-Rong Wen and Xiaojie Yuan, "Evaluating the Effectiveness of Personalized Web Search", IEEE Transactions on Knowledge And Data Engineering, Vol. 21, No. 8, August 2009.


