Two phase methodology for optimized web page recommendation

Manikandan.R, Research Scholar- Anna University
Saravanan V, Professor-Sri Venkateswara College of Computer Applications and Management

***Abstract*- In the digital era, An end user mostly depend on the WWW for information,but the search engines which are used frequently often retrieves a large number of results many of which are not always relevant to the users' need. Web Logs are repositories that store vital information of user activities and navigation. Suitable data mining techniques when applied to it can definitely improve the performance of search engines, making the user to attain his goal of getting the relevant information. The web search can be optimized to provide the user with exact web pages from where he can obtain the information of his search goal. In this paper, we propose a web recommendation approach which based on machine learning from web logs and to recommend users with web pages that are of importance to his interests by comparing it with users' historic browsing pattern. Finally, the result is optimized by assigning new ranks to the result pages. The proposed system proves to be efficient in terms of the page ordering and thus reducing the search time.**

Keywords- WWW, Web log; web usage mining; recommendations;Ranking ;Machine Learning

I. INTRODUCTION

As there is an exponential growth of data in internet with time, World Wide Web is never the less a launchpad where users can store, dispatch and retrieve information. Search engines play an important role in finding the needed information to the user by posing a query. Users express their need with a combination of keywords on the interface of search engine. There are many recent notable advances in the Search engine[1] technologies but in many occassions, users are provided with un-desired and irrelevant pages as the top results for their searches. This is primarily due to the lack of the user knowledge while framing the queries.

Extracting information from an online navigation is a trivial task as the naviagation behaviour grows exponentially. Web log files play an important role in achieving this. The log files maintained by the search engines provide an excellent opportunity to understand the interests of the users by maintaining the information within it. One solution to predict the users' navigation behavior is web usage mining (WUM). It has become very difficult nowadays to manage a website, to create a adaptive website for various domains . The task is still more trivial to produce personalized recommendation for web pages based on the user behavior and browsing patterns. A typical application of WUM is recommendation system. The main goal of the recommendation system is to improve website usability.

This paper proposes a web recommendation approach which is based on the machine learning from the web logs stored by the search engine to recommend users with relevant pages. This is achieved by by comparing with user's feedback and by optimizing the search result by re-ranking them. This will result in an enormous deduction search time of listing the web pages of users interest. Rest of the paper is organized as follows: In section II we have discussed the basic terminologies used in the proposed work . In Section III we have elaborately explained the proposed approach of recommendation system in section IV, performance evaluation is done.

Section V concludes our work and introduces the future work.

II. PRELIMINARIES AND RELATED WORK

In this section, a brief description of the terminologies and concepts that are used in the proposed system are given

2.1 Web Log

A web log is a file in which the information are written by a web server each time when a user request a resource from a specific website. The activities of the users in a particular web session are recorded by the web server and are stored as web logs . These log contains information such as IP address, the time on which the request for the page is placed, the URL that was requested, the status and many more. Web log consist of attributes with the data values in the form of records. There are number of ways to use the information available in the log file[6, 7]. The more common usage of the information in the web logs are to gain a strong knowledge on the search process and to improve the performance of web search engines [8]. The log files are also used to discover the semantic relation between the query and the users navigation pattern[2].

2.2 Similarity Function

The process of finding whether the instances of a data are matching with the same real world entity is called as data matching. The similarity between any two vectors is calculated using the value of the cosine value of the angle between them. The similarity between an active browsing session with that of the cluster of aggregate profiles obtained from the web log is measured using the cosine similarity. If an active browsing session is represented as s_j and cluster as c_k, then their similarity can be measured as follows:

$$sim(s_j, c_k) = \frac{\sum_{i=1}^{n} w_{i,j} \bullet w_{i,k}}{\sqrt{\sum_{i=1}^{n} w_{i,j}^2} \bullet \sqrt{\sum_{i=1}^{n} w_{i,k}^2}} \quad \text{---} \quad (1)$$

Where $w_{i,j}$ represents weight of page i in active session j and $w_{i,k}$ represents weight of page i in cluster k. The threshold value is taken as a benchmark and the profiles which have a similarity value greater than the threshold value are segregated as matching clusters.

2.3 Page Ranking

A typical search engine uses algorithm to determine the pages that are relevant to the iser and sorts it accordingly using a ranking algorithm. The commonly used ranking algorithms are HITS[9], WPR[10], and SALSA[4] etc. One of the major reason for the relevant pages not getting listed in the top results is that in most of the approaches, the rank score is independent of users query. The pages displayed are different from the users requirement and desire.The page ranking algoritm is used by google to determine the importance of pages by using its link structure. . A simplified version of PageRank is defined as:

$$PR(u) = (1-d) + d \sum_{v \in B(u)} \frac{PR(v)}{N_v} \quad \text{---} \quad (2)$$

where u represents a web page, B(u) is the set of pages that point to u. *PR(u)* and *PR(v)* are rank scores of page u and v, respectively. *Nv* denotes the number of outgoing links of page v,and d is called damping factor.

III. PROPOSED APPROACH

The proposed architecture is divided into two main phases the rear phase and front phase. In the rear phase, there are main two modules: Data pre-processing and sequential pattern mining. The block diagram of recommendation system is given below:

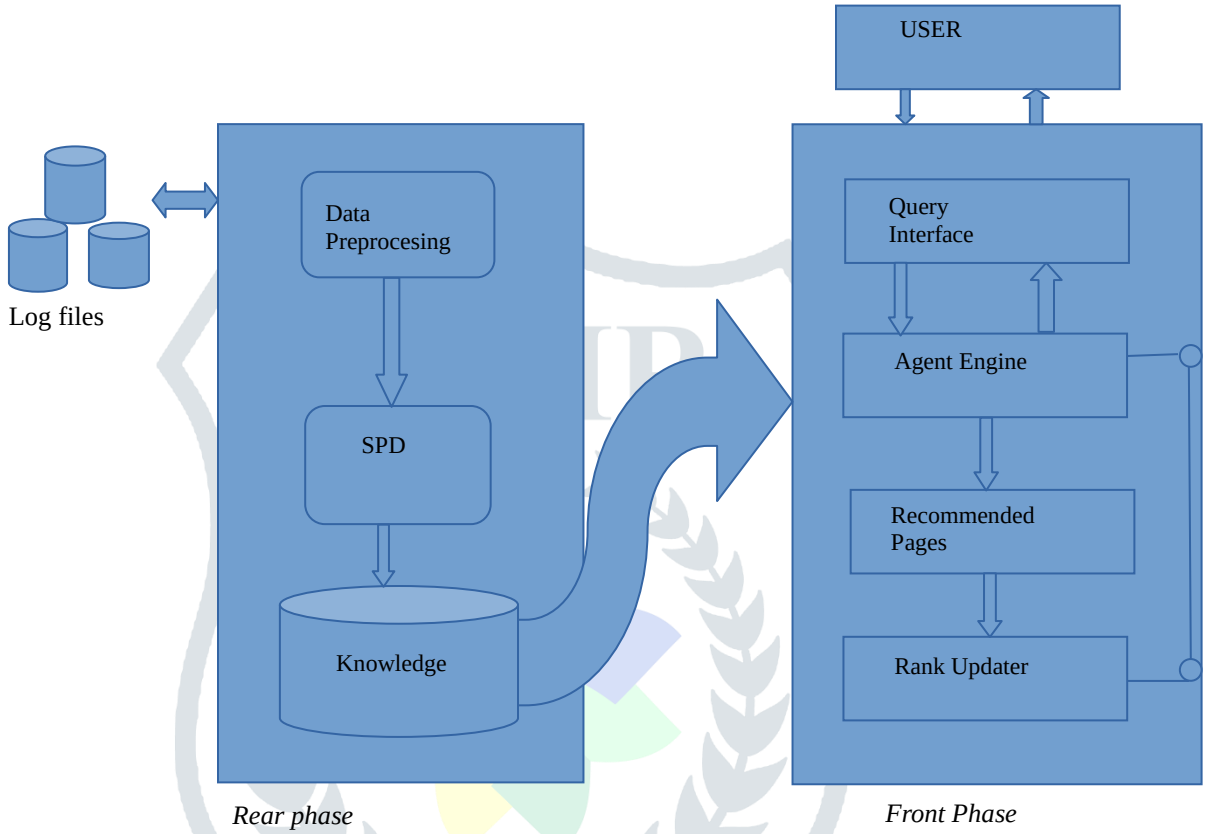


Figure 1 : Architecture of Proposed System

3.1 Rear Phase Architecture:

Rear Phase consists of two modules i)Data preprocessing and ii)Sequential pattern mining. The navigation session of the user can be obtained from Data Preprocessing. The data from the web log will be fo different types as each server has its own server setting parameters which share some information. Preprocessing is a primitive data mining technique which is used here to reformat or to bring in a common form for representing data from different server logs. This is important in obtaining user browser sessions' information. The major activities of the users are registered as a log file by the server which includes time of request, IP, URL ,Status code, referrer ,etc.

3.1.1 Data Preprocessing:

The preprocessing of web logs is complex and time consuming and it is done using following steps:

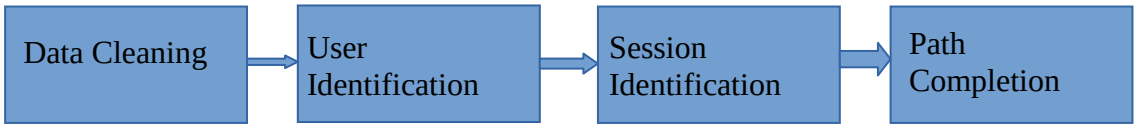


Fig. 2 Process of Data Pre-processing

a)Data Cleaning: -The purpose of data cleaning is to eliminate irrelevant items and these kind of techniques are important for any type of web log analysis. Following kind of records are unnecessary and should be removed.All logs entries with file name suffixes such as GIF, JPEG, gif, jpeg, JPG & map.

b) User identification: -User identification is used to identify who access website and which pages are accessed. The different IP addresses distinguish different users. If the IP addresses are same, different browsers and operating system indicate different users which can be found by client IP address. A method called navigation pattern is used to identify user automatically[3].

c) Session identification and reconstruction: -It involves two steps
1. Identifying the different user session from usually very poor information available in log files.
2. Reconstructing the user navigation path within the identify session.

d) Path completion: -Path completion should be used acquiring the complete user access path. If a page request is made that is not directly linked to the last page a user requested, the log can be checked to see what page request came from. If the page is in user's
recent request history, it means user called up cached version of pages with "back" button until a new page was requested.

3.1.2 Sequential Pattern Mining:

The next step in rear end is to determine the sequential patterns in each cluster[5]. Let W be the set of unique access events, which represents web resources accessed by users which may be web pages, URLs, query topics or categories. A web access sequence S = W1W2---Wn (Wi∈W) for 1≤i≤n is an ordered sequence of access events, and |S| = n is called the length of the web access sequence. Although access events can be repeated in a web access sequence, any web access sequence S can get a support of at most one from each web access sequence Si. The support of web access sequence S in the database is the total number of unique web access sequences that contains S and we represent sup(S) = |{Si | S □□Si |. A web access sequence S is called a sequential web access pattern, if sup(S) ≥ MinSup, where MinSup is a given support threshold. An access pattern Wi∈W is called a frequent pattern, if sup(Wi) ≥ MinSup. Otherwise, it is called an infrequent pattern. Only frequent patterns of events in the current access sequence are considered for generating the next candidate sequence. For all patterns P in the candidate set with length n, all URL sequences are processed once and the count is incremented for each detected pattern in the candidate set. At each iteration, candidate n sequences whose support is less than support threshold are eliminated by the module.

3.2 Front end phase of architecture:

In the front end phase, URL request of the user is processed by search engine and captures the recommended list of web pages relevant to user query and then rank updating algorithm is applied on them.

Step1: Query Matching Algorithm

a. User submits the query on query interface.

b. The query terms are compared to the already existing index document depository from the knowledge base.

c. Apply similarity function(1) to the query terms with discovered aggregate documents and documents having a similarity greater than a threshold are selected.

d. These matching documents can be used for recommendation list.

Hence, recommendation list as compared to user's query are captured.

Step2: Re-ranking Algorithm

a. For the given user query, a set of matched documents are returned by the matching query algorithm.

b. The sequential patterns of the concerned cluster are retrieved from the local depository of the sequential pattern generator.

c. For every page x in the sequence pattern, calculate its value which is based on order in which that has been accessed and important for the user. This is calculated as:

val(x)= ln(depth)/level(x) ----- (3)

Here depth is the effective depth of the sequential pattern sequence in which page x lies and level(x) is the level of page x in the sequence.

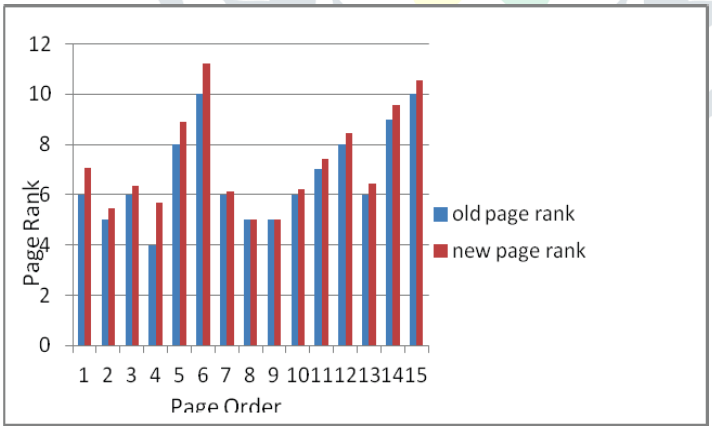
d. The new improved rank of page x is calculated as:

Improved_Rank(x)=Rank(x)+val(x) ---- (4)

Hence, the popular and relevant pages gain the upwards position in the recommendation list.

IV. PERFORMANCE EVALUATION

It may be observed that the pages which are frequently accessed by users have a change in their rank values. Some of the pages have same rank as before. It can be evaluated from the results that the ranking of many web pages have been modified. Thus, more relevant Web pages occupy the top positions in the result list as shown below according to the above implementation.



V. CONCLUSION

A two level architecture is proposed in this paper to obtain optimized web pages for recommendation . A query matching algorithm and Re-Ranking algorithm are proposed for an efficient web search. The use of users'feedback and web log mining have made a significant improvement in the results. The result show that the proposed system improves the relevancy of the pages and thus reduce the time user spends in seeking the required information. It is also shown that the rank of the page is enhanced from the previous methods significantly. In future, The same algorithm can be improved to make it efficient when applied to cross domains.

References :

[1] A. Arasu, J. Cho, H. Garcia-Molina, A. Paepcke, and S.Raghavan, "Searching the Web," ACM Transactions on Internet Technology, Vol. 1, No. 1, pp. 97-101, 2001
[2] Edgar Meij, Marc Bron, Bouke Huurnink, Laura Hollink, and Maarten de Rijke. Learning semantic query suggestions. In 8th International Semantic Web Conference (ISWC 2009). Springer, October 2009..
[3] J.Ben Schafer, Joseph Konstan, John Riedl "Recommender Systems in E-Commerce" GroupLens Research Project Department of Computer Science and Engineering, University of Minnesota.
[4] R. Lempel and S. Moran, "SALSA: The Stochastic Approach for Link-Structure Analysis". ACM Transactions on Information Systems, 19(2), Apr 2001, pp: 131–160.
[5] Murat Ali Bayir, Ismail H. Toroslu, Ahmet Cosar. Performance Comparison of Pattern Discovery Methods on Web Log Data. Proceedings of AICCSA, 2006, pp: 445-451.
[6] K. Hofmann, M. de Rijke, B. Huurnink, E. Meij. A Semantic Perspective on Query Log Analysis, In Working notes for the CLEF 2009 Workshop, Cortu, Greece.
[7] H. Ma, H. Yang, I. King, and M. R. Lyu. Learning latent semantic relations from clickthrough data for query suggestion. In CIKM '08: Proceeding of the 17th ACM conference on Information and knowledge management, pages 709–718, New York, NY, USA, 2008. ACM.
[8] Thorsten Joachims. Optimizing search engines using clickthrough data. : Proceedings of the 8th ACM SIGKDD.
[9] S. Chakrabarti, B. E. Dom, S. R. Kumar, P. Raghavan, S. Rajagopalan, A. Tomkins, D. Gibson, and J. Kleinberg. Mining the Web's link structure, 32(8):60-67, 1999.
[10] Wenpu Xing and Ali Ghorbani, "Weighted PageRank Algorithm", Proc. of the 2nd Annual Conference on communi