IMPROVED SEARCH RESULTS USING FEEDBACK SESSIONS

Kota Vamsee Krishna[#1], Smita Deshmukh[#2]
#Information Technology, Terna Engineering College, Mumbai University
Sector-22, Nerul, Navi Mumbai, Maharashtra, India
[1]vamsee123in@yahoo.com
[2]deshmukhsmita17@yahoo.com

Abstract— **Before submitting any query to a search engine, every user has a specific goal in mind but that goal is not known to the search engine. Based on the query related matching information which is currently available in the database, search engine will display the results and user will have to scan through them to find out the website of his interest everytime. From user experience perspective, this is a very time consuming and recurring activity. User would be happy if scanning several results to find out required information for a particular query can be reduced to a one time activity. User would practically benefit if search results could be displayed in categories (such as Business, Science and Technology, Sports etc.) with manual rating so that user defined highest rated URLs are displayed at the top which no search engine would do. It will help user to save effort and time in fetching required information. This can be achieved by implementing a desktop-based application. This paper is about a desktop-based application which will fetch search results from Google and Microsoft Bing together in one go and will infer user search goals. Users will have to register themselves and create user id and password. Considering one login/logout as a single session, usage logs will be captured and feedback sessions will be generated to restructure and optimize display of search results. The performance of this application will be evaluated using Classified Average Precision (CAP) factor.**

Keywords— **User search goals, feedback sessions, manual rating, restructuring search results, classified average precision.**

I. INTRODUCTION
In today's world, if anyone needs any information then the first thing that they do is to search for it in any of the popular search engines using Internet. Some of the popular search engines are Google, Microsoft Bing etc.. Let us call the information that user is looking for as a 'Query'. Before entering the query in search engine, every user has a goal (i.e. specific information) in mind that they want to achieve. But the search engine does not know what exactly the user is looking for, so it will display all the available URLs belonging to various domains without any categorization. The user has to scan through all the displayed URLs one-by-one until the desired URL is identified. Practically every user wants the required/relevant/wanted/needed information to be quickly displayed to save time and this would be possible only if there is an improvement in the display of search results. This gives rise to the need for developing a desktop based application with categorization and user defined manual rating features which will analyse usage logs, produce feedback sessions and restructure the search results whenever user submits the same query next time and evaluate the performance of the restructured results using Classified Average Precision(CAP).

II. RELATED WORK
The authors of reference paper [1], propose a unique approach to conclude user search goals by a deep analysis of search engine query logs. The work in reference paper [2], talks about text classification algorithms used to automatically classify random search results into an existing category structure as a continuous on-going activity. Organizing search results will allow a user to concentrate on URLs in the user required categories. Further, the reference paper [3] discusses about organization of the web search results into a cluster and thus facilitate a user in quick browsing of the search results. The reference paper [4] highlights regarding effective organizing of search results as a critical activity so as to improvise the utility of any search engine. To navigate through relevant documents easily, clustered search results prove to be efficient. Displaying the relevant documents in the order starting from most relevant to least relevant is done by a good information retrieval system and this is explained in reference paper [5]. The goal is to develop a method that utilizes click-through data for training, namely the query-log of the search engine in connection with the log of links the users clicked on in the presented ranking. As per reference paper [6] depending on the clicks relative preferences are obtained which are averagely accurate. The work in reference paper [7] presents a method which is based on query clustering process. It facilitates grouping of semantic similar queries which could be identified. Referring to the work in reference paper no. [8], it proposes and evaluates a method for auto-segmentation of user's query streams into distinct units. Reference paper [9] states that query substitution is generation of a new query inorder to replace the user's old query. The work in reference paper [10] presents increase of precision retrieval, the new search engines input manually verified answers to frequently asked queries.
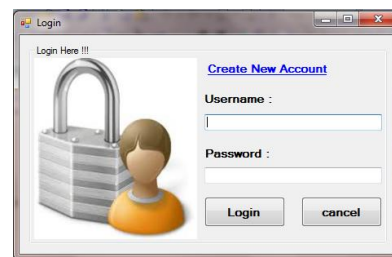
III. FRAMEWORK
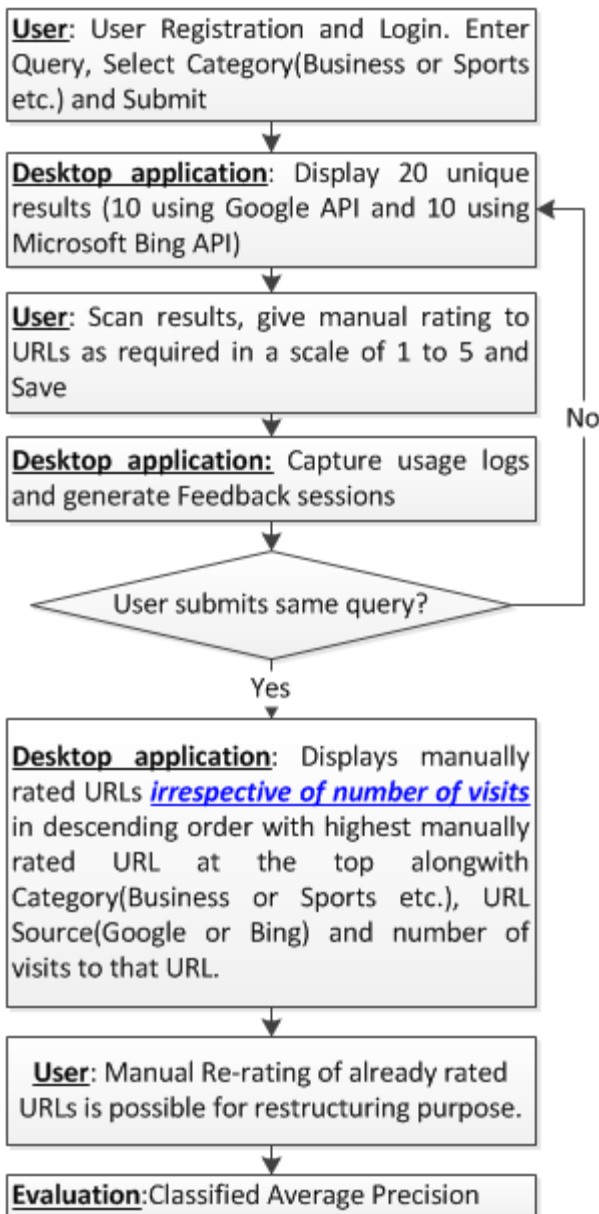The framework is illustrated in below diagram.

Fig1: Framework

IV. EVALUATION

Evaluation through Classified Average Precision(CAP) will prove that our method of inferring user search goals is better than old method explained in reference paper [6].

A. Average Precision. A possible evaluation criterion is the average precision (AP) which evaluates according to user implicit feedbacks.

$$AP = (1/N^+) \left[\sum_{r=1}^{N} rel(r).(R_r/r) \right] \qquad (1)$$

where N^+ is the number of manually rated URLs in the retrieved ones, r is the rank, N is the total number of retrieved documents, *rel()* is a binary function on the relevance of a given rank, and R_r is the number of relevant retrieved documents of rank r or

less. However, AP is not suitable for evaluating the restructured results. Hence, we use Voted Average Precision.

B. Voted Average Precision (VAP). Voted Average Precision (VAP) is the average precision of the class including more manually rated URLs namely votes. If the number of the manually rated URLs in two classes are the same, the bigger AP is selected as the VAP.

However, VAP is still an unsatisfactory criterion. To overcome the limitations of VAP, CAP is proposed.

C. Classified Average Precision (CAP). CAP selects the Average Precision of the class that user is interested in (i.e., with the most manually rated URLs/votes) and takes the risk of wrong classification into account. The influence of Risk on CAP is adjusted, which can be learned from training data. Finally, we utilize CAP to evaluate the performance of restructuring search results.

The CAP depends on both Risk and VAP.

$$CAP = VAP \times (1 - Risk)^\gamma \qquad (2)$$

γ is used to adjust the influence of *Risk* on CAP, which can be learned from training data.

$$Risk = \sum_{i,j=1(i<j)}^{m} d_{ij} / C_m^2 \qquad (3)$$

It calculates the normalized number of manually rated URL pairs that are not in the same class, where m is the number of the clicked URLs. If the pair of the ith manually rated URL and the jth manually rated URL are not categorized into one class, d_{ij} will be 1; otherwise, it will be 0.

$C_m^2 = m(m-1)/2$ is the total number of the manually rated URL pairs.

VII. EXPERIMENTS

To implement the desktop based application, Google API and Microsoft Bing API was downloaded and added to the reference library so that application could fetch URLs from both search engines.

As a first step, user has to register by clicking on 'Create New Account', fill-in all the details and thus create a user id and password.

Fig.2: Login page

Fig.3: Registration page

Using User ID and Password, user has to login to the 'Desktop Application'. Then User has to enter Query, select 'Area of Interest' i.e. Category and click on 'Search'.
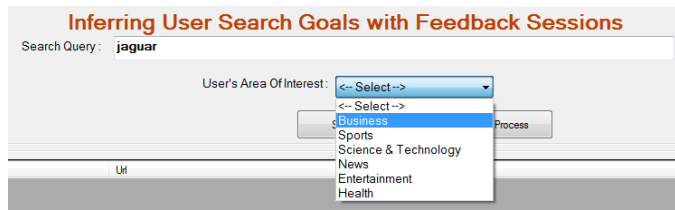

Fig.4: Query submission

20 unique results(10 from Google API and 10 from Microsoft Bing API) will be displayed as shown below. The desktop application displays results in the same sequence as displayed in Google and Bing which will facilitate display of popular URLs too. Since we are using free APIs of Google and Bing hence we have restricted the display of results to 20 unique results per search. User can click on URLs to access respective webpages and can give manual rating in a scale of 1 to 5 and click on 'Save'.
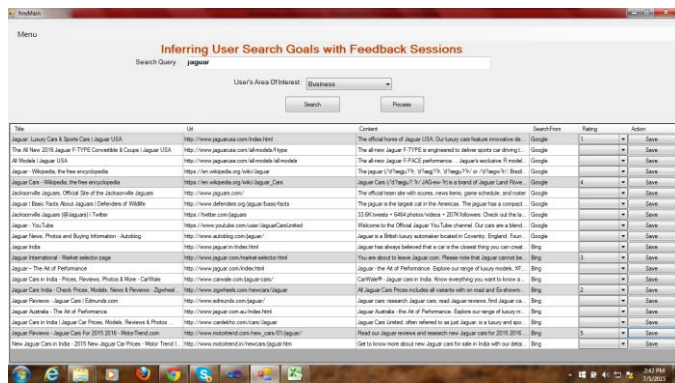
Fig.5: Display of original search results

As shown below 'Session log' will contain 'Usage logs'. Navigation path is Menu-> Session Log.
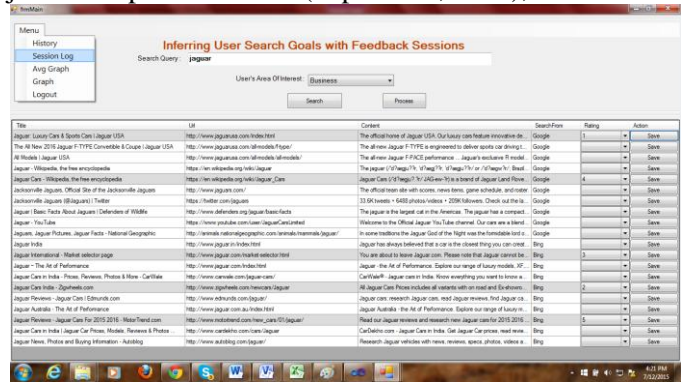
Fig.6: Navigation path-Session Log

Under 'Session log', usage logs can be viewed as shown below. In 'Session log', we can see query submitted, how many URLs were displayed, which URLs were manually rated by user and which URLs were not manually rated. Those which are manually rated by User indicate that they are the 'User Search Goals'.
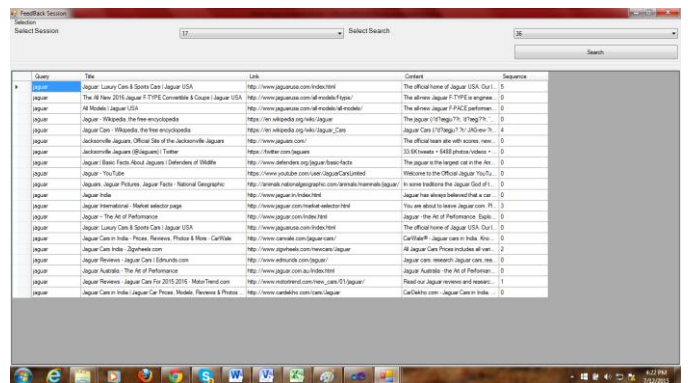
Fig.7: Session Log

Based on 'Session log', the desktop application generates 'Feedback session'. Navigation path is Menu->History.
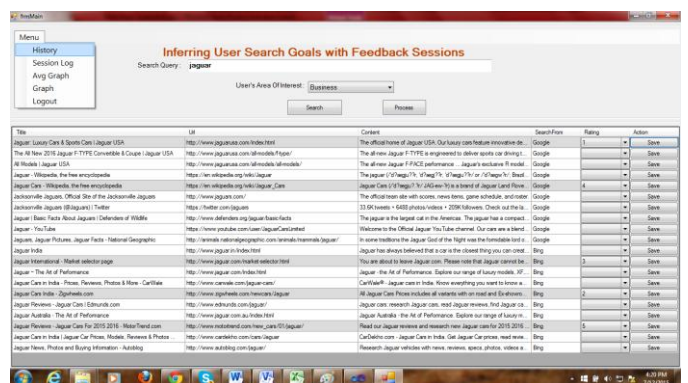
Fig.8: Navigation path-Feedback session

As shown below, 'Feedback session' contains only those URLs which were manually rated by User because they are the 'User Search goals'.

Fig.9: Feedback session

Next time if User enters the same query, user can directly click on 'Process' tab to view those which were manually rated earlier because they are those websites which contain user required information.

As shown below, User entered the same query 'Jaguar' and clicked on 'Process'.

Fig.10: Re-submission of same query

Upon clicking on 'Process' tab, 'User search goals' will be displayed as shown below.

Fig.11: Manually rated results

User can also re-rate above results by submitting the same query again and then user can manually change the rating. Re-Rating can be done irrespective of the number of visits to a particular URL. It will help user to view required information irrespective of number of visits to the URL because that URL is 'User search goal'.

For example, in above screenshot, we can see that http://www.motortrend.com/new_cars/01/jaguar/ was manually rated by User as '5' hence its displayed at the top and http://www.jaguar.com/market-selector.html was manually rated by User as '3' hence its displayed accordingly. User can change it.

As per below screenshot, User submits the same query again and gives rating to http://www.motortrend.com/new_cars/01/jaguar/ as '3' and rates http://www.jaguar.com/market-selector.html as '5'.

Fig.12: Manual Re-rating

Now, when user clicks on 'Process' tab, rating gets changed.

Fig.13: Manual Re-rated results

As per above screenshot, we can see that http://www.motortrend.com/new_cars/01/jaguar/ was earlier manually rated by User as '5' and then changed to '3' is now displayed as '3'. And http://www.jaguar.com/market-selector.html which was earlier manually rated by User as '3' and then changed to '5' is now displayed as '5'.

Thus, User can re-structure the search results.

CAP value for our method considers only the manually rated URLs whereas the old method[6] considers all the 20 displayed URLs. CAP value Graph for every query can be viewed under 'Graph' section. Navigation path is Menu->Graph.

Fig.14: Navigation path-CAP Graph

As per below CAP graph, we can see that CAP of our method is always better than old method.
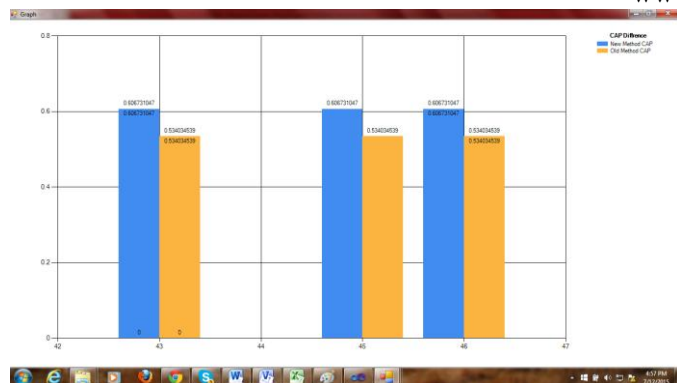
Fig.15: CAP Graph per query of individual user

Average of all graphs can be viewed under Menu->Average graph.
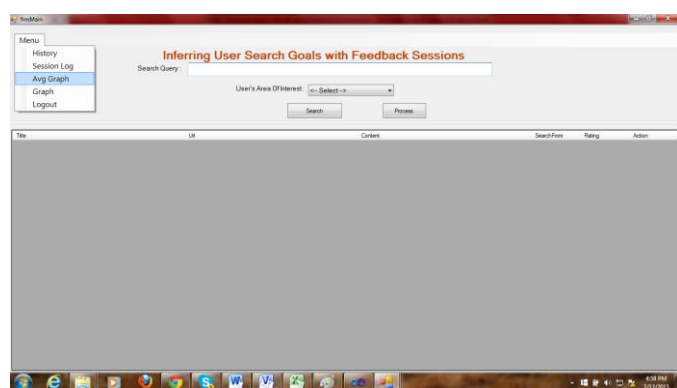

Fig.16: Navigation path-Average CAP Graph

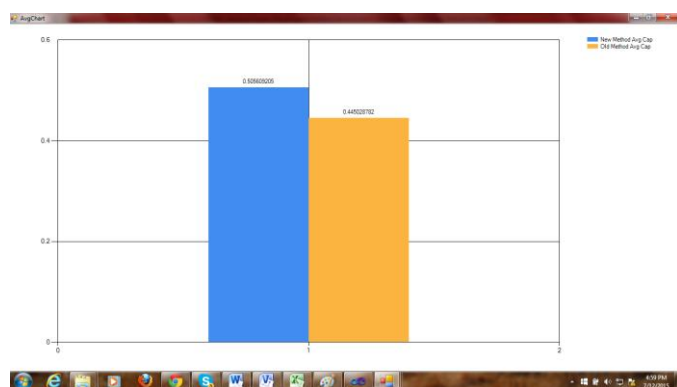Average graph considers all the queries submitted by individual user.

Fig.17: Average CAP Graph for all queries of individual user

Hence, CAP of our method is always better than old method.

VIII. CONCLUSION

In reality, our approach of manual rating is far better than search engine ranking because search engines such as Google do ranking based on usage. So, in case of Google, if user wants a particular URL to be displayed at the top then user will have to visit that URL maximum number of times which is an additional effort. Whereas in our desktop-application, manual rating will restructure the results as per User requirement irrespective of number of visits. Thus, this desktop application can infer user search goals and save user effort and time. Hence, users can find what they want conveniently.

REFERENCES

[1] Zheng Lu, Hongyuan Zha, Xiaokang Yang, Weiyao Lin and Zhaohui Zheng, "*A New Algorithm for Inferring User Search Goals with Feedback Sessions*", IEEE transactions on Knowledge and Data Engineering, Vol.25, no.3, pp. 502-513, 2013.

[2] H. Chen and S. Dumais, "*Bringing Order to the Web: Automatically Categorizing Search Results,*" Proc. SIGCHI Conf. Human Factors in Computing Systems (SIGCHI '00), pp. 145-152, 2000.

[3] H.-J Zeng, Q.-C He, Z. Chen, W.-Y Ma, and J. Ma, "Learning to Cluster Web Search Results," Proc. 27th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR '04), pp. 210-217, 2004.

[4] X. Wang and C.-X Zhai, "*Learn from Web Search Logs to Organize Search Results,*" Proc. 30th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR '07), pp. 87-94, 2007.

[5] T. Joachims, "*Optimizing Search Engines Using Clickthrough Data,*" Proc. Eighth ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (SIGKDD '02), pp. 133-142, 2002.

[6] T. Joachims, L. Granka, B. Pang, H. Hembrooke, and G. Gay, "*Accurately Interpreting Clickthrough Data as Implicit Feedback*," Proc. 28th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR '05), pp. 154-161, 2005.

[7] R. Baeza-Yates, C. Hurtado, and M. Mendoza, "*Query Recommendation Using Query Logs in Search Engines,*" Proc. Int'l Conf. Current Trends in Database Technology (EDBT '04), pp. 588-596, 2004.

[8] R. Jones and K.L. Klinkner, "*Beyond the Session Timeout: Automatic Hierarchical Segmentation of Search Topics in Query Logs,*" Proc. 17th ACM Conf. Information and Knowledge Management (CIKM '08), pp. 699-708, 2008.

[9] R. Jones, B. Rey, O. Madani, and W. Greiner, "*Generating Query Substitutions,*" Proc. 15th Int'l Conf. World Wide Web (WWW '06), pp. 387-396, 2006.

[10] J.-R Wen, J.-Y Nie, and H.-J Zhang, "*Clustering User Queries of a Search Engine,*" Proc. Tenth Int'l Conf. World Wide Web (WWW '01), pp. 162-168, 2001.