*International Journal of Research in Advent Technology, Vol.2, No.11, November2014*
*E-ISSN: 2321-9637*

# Efficient Web Based Text Query Results Refining Mechanism Using Relevance Feedback

Mounika Komirelli[1], M. Raja Krishna Kumar[2]
*Department of CSE [1, 2], Geethanjali College of Engineering and Technology [1, 2]*
*mounikareddy1214@gmail.com[1], munagala_rajakrishnakumar@rediffmail.com[2]*

**Abstract:** The World Wide Web has become one of the important information resources for us. Search engines available now a day are giving results based on individual user's search history which is centralized to that particular user. Here we are trying to refine the search results irrespective of user by considering the user feedback. The refinement search results can have lot of advantages in improving search engine relevance and user experience. We can reconstruct web search results with same search query. Results are refined in such a way that they are displayed in the order of relevancy.

**Index Terms:** Relevance feedback; Refinement; Click through data.

## 1. INTRODUCTION

World Wide Web is a very important information tool which survives many of the informational needs of the users. It provides us with the rich form of data which includes text, images, graphics, etc.

In reference paper [2] User queries can be categorized into two basic types they are "navigational" and "informational".

User keeps on searching on web but he/she often fail to get the appropriate results as per the relevancy. Order of display of results will be affected by various issues like commercial, hit count, user ratings etc. In this paper we are refining the search results display order based on relevance feedback.

***1.1 Refinement:*** This is the process of producing better results than available. Here we refine search results so as to satisfy the informational needs of the user. Refinement is subjected to change over time based on the relevance feedback. Refinement is used in increasing relevancy and efficiency of search engine. We can restructure web search results. Refinement is very useful in re-ranking of web search results.

***1.2 Relevance Feedback:*** We have 3 different types of feedbacks relevance feedback is the feature of informational retrieval systems. It is the process of taking results which are initially retrieved and using them further in improving the future results for further queries. Relevance feedback helps in automatic analysis.

## 2. PRESENT APPROACH

Refinement of results is involved in 6 major steps. Here results will be refined and displayed irrespective of user so that results can be generalized.

If we consider GOOGLE as example it gives results with respect to individual user base on his login history. It means user get more relevant results when he/she login into GMAIL account [3].

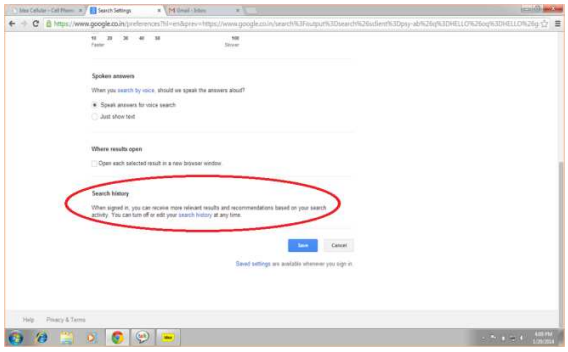


Fig 1: figure showing GOOGLE search history notice

In fig(1) GOOGLE itself saying that when signed in you can receive more relevant results and recommendations based on your search activity. It means we can get more relevant results if we signed in otherwise we will get routine results. Here the relevancy is depended on user.

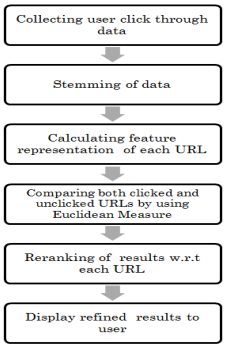


Fig 2: Overview of the work

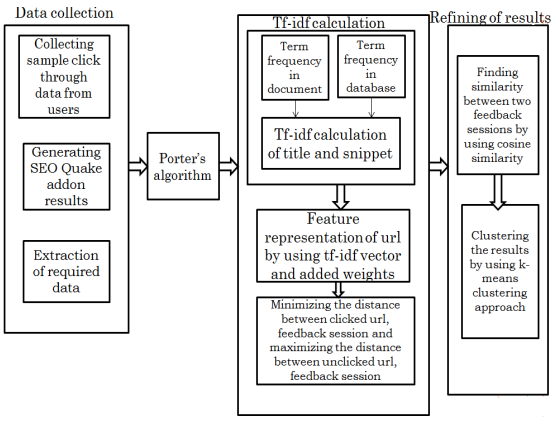


Fig 3: Architecture of the system

### *2.1 Collecting User Click Through Data*

In this the user will post some query on the search engine so the web will give some results to the user, in that user will click on some results in that page and gets some information. So here we are providing an excel sheet to the user where we will give some fields like serial number, posted query, clicked url, position of the url, required information is formed or not. We will gather this information from users.

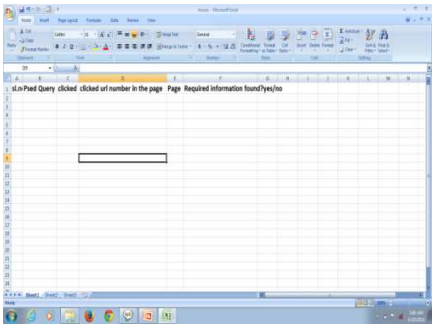

Figure 4: Figure showing sample click through data collection

In figure 4 posed query represents query posted by user, clicked URL number in the page means position of the user clicked URL in the page, page represents in which page clicked URL has appeared, as of now we are just dealing with first page results and required information found? Represents whether the user has found required information or not if found, then the value will be yes else no.

From the data we have collected we have analyzed that most of the URL are displayed in the first position but not providing the required information to the user. Hence we are not considering these URL's in the click through data.

### *2.2 SEO Quake Add-on[6]*

On installing the add-on, whenever we pose a query to search engine a control panel will appear on the top right corner of the window in that an option called SERP control panel in that we have another option called EXPORT TO CSV this option will export top ten results of GOOGLE in comma separated file of excel sheet. When we get that comma separated file we will provided with following fields they are position, URL, title, description, page authority etc. From this we extracting only four fields they are position, URL, title, description, page authority. From this we use description which is also called as snippet use throughout the document. In this document all fields will be there but we require only four fields.

### *2.3 Porters Stemmer*

In our work porters stemmer [4] has been used to stem the data we have collected. Stemmer has been helped in standardizing the text as per our requirement. Extra endings have been removed and root of the word has been identified.

Table 1: Sample stemming data

| RAW DATA | STEMMED DATA |
|---|---|
| Bank Notification Recruitment for 206 Officer Office Assistant post 2014 at the Online CWE for RRB conduct by IBPS during September October 2013 Sarva UP Gramin Bank Dena Gujarat Gramin Bank | bank 55otify recruit for 206 officer office assistant post 2014 at the online cwe for rrb conduct by ibps dure 55otify5555r 55otify55 2013 sarva up gramin bank dena 55otify55 gramin bank |

In table 1 we can observe data has been stemmed words like "notification" has been stemmed to "otify", "Recruitment" has been stemmed to "recruit" and all the text is converted to lower case.

### *2.4 Stop List Algorithm*

Stop list algorithm [4] helped us in removal of words which doesn't have any weight and which are not informative. Stop list is nothing but words list which are to be stopped, like prepositions.

Table 2: Stoplist applied data on stemmed data

| STEMMED DATA | DATA AFTER APPLYING STOP LIST |
|---|---|
| Bank 55otify recruit for 206 officer office assistant post 2014 at the online cwe for rrb conduct by ibps dure september october 2013 sarva up gramin bank dena gujarat gramin bank | bank notifi recruit 206 officer office assistant post 2014 online cwe for rrb conduct by ibps dure september october 2013 sarva up gramin bank dena gujarat gramin bank |

In table 2 we can observe stop words removed in 2nd column. So that some of the words which don't have much importance will be eliminated.

### *2.5 Tf-Idf Vector Calculation*

As per reference paper [1] for each and every URL we have title and snippet. Title and snippet will be having some words and the length of the title and snippet are not same. A word table will be maintaining with wordID and words. Words in the word table are unique. To calculate tf-idf vector for title and snippet we need to equalize the length of title and snippet. We replace words of title and snippet with wordID from the word table. After replacing the words with wordID we will take each and every word from the word table and compare it with word id of title and snippet respectively, if the word is present in the title then the value will be set to tf-idf value of the word otherwise 0.

$$\mathbf{T_{ui} = [t_{w1}, t_{w2}, t_{w3} ,\ldots\ldots\ldots t_{wn}]}$$
$$\mathbf{S_{ui} = [s_{w1}, s_{w2}, s_{w3} ,\ldots\ldots\ldots s_{wn}]}$$

$T_{ui}$ and $S_{ui}$ are tf-idf vectors if title and snippet respectively.

We will calculate tf-idf values by using following formula [9]

$$tf(t,d) = 0.5 + \frac{0.5 * f(t,d)}{\max\{f(w,d) : w \in d\}} \text{ (i)}$$

$$idf(t,d) = \log \frac{N}{\{d \in D : t \in d\}} \text{ (ii)}$$

***Tf-Idf(t,d) = tf(t,d )* idf(t,d) (iii)***

In (i) we are calculating term frequency f(t,d) represents frequency of term t in the document d and max{f(w,d)} represents word w which has maximum count in the document.

In (ii) we are calculating idf value N is total no. of documents in the database and d represents total number of documents which has the term t.

After obtaining tf and idf values in (iii) we can obtain tf-idf vector by multiplying tf idf values from (i),(ii).

By using the formula of tf-idf vector we can calculate tf-idf vector for title and snippet.

Table 3: Tf-idf vectors of title and snippet.

| Title | Tf-idf vector of title | Snippet | Tf-idf vector of snippet |
|---|---|---|---|
| code andhra bank kookapet branch ranga | [1.95,0.97, 0.975,0.97, 0.97,0.97,0.97,0.97,0.975,0.975,0.975,0.975,1.3,0.975,0.9 | com andhra pradesh ranga reddi district cachedsimilarifsc code detail and | 1.95,0.975,0.975,0.975,0.975,0.975,0.975,0.975,1.1375,1.1375,0.975,1.4625,0.975,1.1375,1.1375,0.975,1.1375, |

| | | | |
|---|---|---|---|
| | 75,1.3,0.97<br>5,1.3,0.975,<br>0.975,0.97,<br>0.975,0.975<br>,0.975,1.3,0<br>.97,0.975,0.<br>975,0.975,0<br>.975,0.975,<br>0.975,0.975<br>,0.975,0.97,<br>0.975,0.975<br>,0.975,0.97,<br>0.975,0.97,<br>0.975,0.97,<br>0.975,1.3,0.<br>97,0.975,0.<br>97,0.975,0.<br>975,0.975,1<br>.3,0.975,0.9<br>75,1.3,0.97<br>5,0.97,0.97<br>5,0.97,0.97<br>5,0.975,0.9<br>75,0.975] | kookapet<br>branch<br>inform<br>andhra bank<br>kookapet<br>locat andhra<br>pradesh state<br>ranga reddi<br>district<br>kokapet | 0.975,0.975,0.97<br>5,0.975,0.975,0.<br>975,1.1375,0.97<br>5,0.975,0.975,1.<br>1375,0.975,0.97<br>5,1.3,0.975,0.97<br>5,0.975,0.975,0.<br>975,0.975,0.975,<br>0.975,0.975,1.13<br>75,1.1375,0.975,<br>1.3,0.975,1.1375<br>,0.975,0.975,0.9<br>75,0.975,0.975,0<br>.975,1.3,1.3,1.13<br>75,1.1375,0.975,<br>0.975,0.975,1.13<br>75,0.975,0.975] |

In the table 3 we have title, snippet and their corresponding tf-idf vectors title and snippet vector lengths are same and equal to total number of words present in the word table.

### *2.5 Feature Vector Representation*

In Reference paper [1] Calculation or feature vector will be done by using added weights

$$\mathbf{F_{ui} = w_t T_{ui} + w_s S_{ui}}$$

$$\mathbf{= [f_{w1}, f_{w2}, \ldots\ldots f_{wn}]^T}$$

In the above formula $w_t$ and $w_s$ are added weights. In [1] the weights are set to be 2 and 1 respectively. Weight of the title is more than the weight of the snippet because most of the users concentrate much on title rather than snippet.

Finally we obtain a vector for every URL. Here each term indicates the importance of the term in $i^{th}$ URL.

By using above formula feature vectors for the sample data are calculated as shown in Table 3

Table 4: Feature vector for the data in Table 3

| | |
|---|---|
| Feature vector | [5.85,2.92,2.92,2.92,2.92,2.92,2.92,2.92<br>,2.92,2.92,3.08,2.92,4.06,2.92,3.087,3.7<br>3,2.92,3.73,2.92,2.92,2.92,2.92,2.92,2.9<br>2,3.73,2.92,2.92,2.92,3.08,2.92,2.92,3.2<br>5,2.92,2.92,2.92,2.92,0.92,2.92,2.92,2.9<br>2,2.92,3.085,3.087,2.92,3.9,2.92,3.08,2.<br>92,2.92,2.92,2.92,3.57,2.92,3.25,3.9,3.0<br>87,3.08,2.92,2.92,2.92,3.087,2.92,2.92] |

### *2.6 Refining the Search Results*

In reference paper [1] with the obtained pseudo documents they have inferred user search goals and goal texts. And to compare two pseudo documents they have used cosine similarity.

$$Sim_{i,j} = COS(F_{fsi}, F_{fsj})$$

$$\frac{Ffsi \cdot Ffsj}{|Ffsi||Ffsj|}$$

In our approach Euclidean distance [10] measure is used in two stages. First, to calculate distance between feature vectors of each clicked URL and feature vectors of URLs in original results. Second, between each clicked URL based re-ranking order and order of original search results and between each clicked URL based re-ranked order and final re-ranked results.

Euclidean distance of two vectors is calculated by using formula

$$\sqrt{\sum (X_i - X_j)^2}_{[5].}$$

In the above formula $X_i$, $X_j$ are feature vectors of two URLs. By applying above formula we can obtain a matrix of Euclidean distances w.r.t URL where the values of diagonal are 0. After obtaining the distances we will sort and calculate average of distances of URLs w.r.t to clicked URL. Finally we sort the URLs w.r.t to average values.

## 3 DISCUSSION OF RESULTS

Results which we have obtained are in a order where the relevancy is more and reduces user's time to choose relevant result among all the results available. We are using m, n values to show our results are better refined than the results which are existed.

Table 5: New sequences corresponding each clicked URL and final sequence.

| Q. Id | Original sequence | New sequence 1 (cl1) | New sequence 2 (cl2) | New sequence3 (cl3) | Final sequence |
|---|---|---|---|---|---|
| Q1 | 1,2,3,4,5,6,7,8,9,10 | 2,4,6,9,3,10,7,8,5,1 | 4,6,2,10,5,9,8,7,3,1 | 6,4,2,3,5,10,7,9,1,8 | 4,6,2,10,3,9,5,7,8,1 |
| Q2 | 1,2,3,4,5,6,7,8,9,10 | 3,2,5,1,4,10,9,8,7,6 | - | - | 3,2,5,14,10,9,8,7,6 |
| Q3 | 1,2,3,4,5,6,7,8,9,10 | 1,3,10,2,7,9,4,5,6,8 | 4,7,3,9,10,5,1,2,6,8 | - | 3,7,1,4,10,9,2,56,8 |
| Q4 | 1,2,3,4,5,6,7,8,9,10 | 1,2,5,3,7,4,6,10,9,8 | 4,5,3,7,2,6,10,1,9,8 | - | 5,2,3,4,7,1,6,10,9,8 |
| Q5 | 1,2,3,4,5,6,7,8,9,10 | 2,1,4,3,10,9,7,8,5,6 | - | - | 2,1,4,3,10,9,7,8,5,6 |
| Q6 | 1,2,3,4,5,6,7,8,9,10 | 1,10,4,9,3,2,7,5,6,8 | 6,5,10,1,4,2,9,8,3,7 | - | 10,1,4,6,5,9,2,3,7,8 |
| Q7 | 1,2,3,4,5,6,7,8,9,10 | 1,3,4,8,5,2,6,9,10,7 | 5,9,1,3,6,2,10,8,4,7 | - | 1,5,3,9,6,2,8,4,10,7 |
| Q8 | 1,2,3,4,5,6,7,8,9,10 | 4,5,2,3,1,6,10,7,9,8 | - | - | 4,5,2,3,1,6,10,7,9,8 |
| Q9 | 1,2,3,4,5,6,7,8,9,10 | 1,5,6,7,8,10,4,9,2,3 | 2,7,6,5,9,1,8,10,3,4 | - | 7,6,5,1,2,8,9,10,4,3 |
| Q10 | 1,2,3,4,5,6,7,8,9,10 | 1,3,10,9,7,2,5,4,8,6 | - | - | 1,3,10,9,7,2,5,4,8,6 |
| Q11 | 1,2,3,4,5,6,7,8,9,10 | 1,9,6,5,8,4,10,7,3,2 | 3,2,5,10,8,6,9,1,4,7 | - | 5,1,6,9,8,3,10,2,4,7 |
| Q12 | 1,2,3,4,5,6,7,8,9,10 | 1,2,4,10,6,8,9,3,7,5 | 5,7,8,6,9,3,10,4,2,1 | - | 8,6,10,4,9,7,2,5,1,3 |
| Q13 | 1,2,3,4,5,6,7,8,9 | 1,2,4,9,5,3,8,7,6 | 2,1,5,4,3,9,8,6,7 | - | 1,2,4,5,9,3,8,6,7 |
| Q14 | 1,2,3,4,5,6,7,8,9,10 | 1,2,9,4,3,7,6,5,10,8 | 7,9,2,4,3,1,5,6,10,8 | - | 9,2,1,7,4,3,6,5,10,8 |
| Q15 | 1,2,3,4,5,6,7,8,9 | 1,7,54,3,8,2,6,9 | 2,6,1,5,9,7,4,3,8 | - | 1,2,5,7,6,4,9,3,8 |
| Q16 | 1,2,3,4,5,6,7,8,9,10 | 1,2,3,10,8,5,7,6,4,9 | 4,3,2,10,8,9,5,7,1,6 | - | 3,2,10,8,4,1,5,7,9,6 |
| Q17 | 1,2,3,4,5,6,7,8,9,10 | 1,2,3,9,7,6,4,8,10,5 | 2,1,3,9,7,6,5,8,4,10 | 3,9,1,7,8,2,6,4,10,5 | 1,3,29,7,6,8,4,5,10 |
| Q18 | 1,2,3,4,5,6,7,8,9,10 | 3,8,2,10,4,5,9,7,6,1 | - | - | 3,8,2,10,4,5,9,7,6,1 |

Table 6. Distance between original sequence and clicked sequence, final sequence

| Q. Id | Cl1-original | Cl2-original | Cl3-original | Final-original |
|---|---|---|---|---|
| Q1 | 1,2,3,5,2,4,0,0,4,9,3 | 3,4,1,6,0,3,1,1,6,9 | 5,2,1,1,0,4,0,1,8,2 | 3,4,1,6,2,3,2,1,1,9 |
| Q2 | 2,0,2,3,1,4,2,0,2,4 | | | 2,0,2,3,1,4,2,0,2,4 |
| Q3 | 0,1,7,2,2,3,3,3,3,3 | 3,5,0,5,5,1,6,6,3,2 | | 2,5,2,0,5,3,5,3,3,2,3 |
| Q4 | 0,0,2,1,2,2,1,2,2 | 1,3,0,3,3,0,3,7,0,2 | | 4,0,0,0,2,5,1,2,0,2 |
| Q5 | 1,1,1,1,5,3,0,0,4,4 | | | 1,1,1,1,5,3,0,0,4,4 |
| Q6 | 0,8,1,5,2,4,0,3,3,2 | 5,3,7,3,1,4,2,0,6,3 | | 9,1,1,2,0,3,5,5,2,2,3 |
| Q7 | 0,1,1,4,0,4,1,1,1,3 | 4,7,2,1,1,4,3,0,5,3 | | 0,3,0,5,4,1,4,1,3 |
| Q8 | 3,3,1,1,4,0,3,1,0,2 | | | 3,3,1,1,4,0,3,1,0,2 |
| Q9 | 0,3,3,3,3,4,3,1,7,7 | 1,5,3,1,4,5,1,2,6,6 | | 6,4,2,3,3,2,2,2,5,7 |
| Q10 | 0,1,7,5,2,4,2,4,1,4 | | | 0,1,7,5,2,4,2,4,1,4 |
| Q11 | 0,7,3,1,3,2,3,1, | 2,0,2,6,3,0,2,7,5,3 | | 4,1,3,5,3,3,3,6,5,3 |

| | | | | |
|---|---|---|---|---|
| | 6,8 | | | |
| Q12 | 0,0,1,6,1,2,2,6,2,5 | 4,5,5,2,4,3,3,4,7,9 | | 7,4,7,0,4,1,5,3,8,7 |
| Q13 | 0,0,1,5,0,3,1,1,3 | 1,1,2,0,2,3,1,2,2 | | 0,0,1,1,4,3,1,2,2 |
| Q14 | 0,0,6,0,2,1,1,3,1,2 | 6,7,1,0,2,5,2,2,1,2 | | 8,0,2,3,1,3,1,3,1,2 |
| Q15 | 0,5,2,0,2,2,5,2,0 | 1,4,2,1,4,1,3,5,1 | | 0,0,2,3,1,2,2,5,1 |
| Q16 | 0,0,0,6,3,1,0,2,5,1 | 3,1,1,6,3,3,2,1,8,4 | | 2,0,7,4,1,5,2,1,0,4 |
| Q17 | 0,0,0,5,2,0,3,0,1,5 | 1,1,0,5,2,0,2,0,5,0 | 2,7,2,3,3,4,1,4,1,5 | 0,1,1,5,2,0,1,4,4,0 |
| Q18 | 2,6,1,6,1,1,2,1,3,9. | | | 2,6,1,6,1,1,2,1,3,9 |

Table 7: Comparison of average of clicked URLs and average of refined sequence

| **Q.id** | **Average($Cl_i$-original) (m)** | **Average (final-original) (n)** | **m/n** |
|---|---|---|---|
| Q1 | 2.91 | 3.2 | 0.909 |
| Q2 | 2 | 2 | 1.000 |
| Q3 | 3.1 | 3 | 1.033 |
| Q4 | 2.57 | 2.28 | 1.127 |
| Q5 | 2 | 2 | 1.000 |
| Q6 | 3.1 | 3 | 1.033 |
| Q7 | 2.3 | 2.2 | 1.045 |
| Q8 | 1.8 | 1.8 | 1.000 |
| Q9 | 3.4 | 3.6 | 0.944 |
| Q10 | 3 | 3 | 1.000 |
| Q11 | 3.2 | 3.6 | 0.889 |
| Q12 | 3.55 | 4.6 | 0.772 |
| Q13 | 1.55 | 1.55 | 1.000 |
| Q14 | 2.2 | 2.4 | 0.917 |
| Q15 | 2.5 | 1.77 | 1.412 |
| Q16 | 2.2 | 2.6 | 0.846 |
| Q17 | 3.2 | 1.8 | 1.778 |
| Q18 | 3.2 | 3.2 | 1.000 |

In the table 7, we have calculated average of ($Cl_i$-original) where i=1,2,..k. k is total number of clicked documents for each query. And we also calculated average of final sequence which is referred as n. finally we have obtained m/n to show how much percentage refined results are better than obtained results.

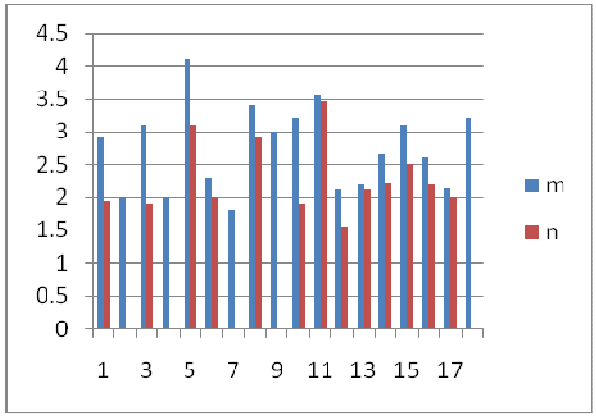


Fig 5: Histogram showing m, n values.

In the figure 5 we can observe two series ‘m’, ‘n’ respectively.

This result is shown for 18 unique queries that means for 25% of data we have collected. m is the value obtained by taking average of differences of clicked URLs with original sequence of results and ‘n’ is the value obtained by calculating average of refined results sequence.

In some cases original results relevancy and refined results relevancy are same, this is happened when we have only one clicked url among all the documents and that too of position one.

In the Existing work, that is in reference 1 they have used CAP method to evaluate their results which consumes much time to calculate. We have achieved better time complexity, as we are using average of values to evaluate the performance. We are just using two parameters ‘m’ and ‘n’. Results which we are giving are refined based on relevance feedback, which is helped in achieving better results. And we are not just only depending on the user clicks we are also considering term frequency and inverse document frequency.

By combining user’s feedback and tf-idf values we are refining the order of results. Refined results are not always constant base on user’s feedback we will be refining the results.

## CONCLUSION

As user's informational needs are increasing day by day he/she is depending on World Wide Web frequently. In order to provide relevant results and to save time of the user we are refining the search results according to relevancy and results are refined irrespective of user. We have taken user's click through data as feedback and with the help of SEO Quake add-on we have collected first page results of GOOGLE and used in our work.

Euclidean distance measures are helped in identifying distance between clicked and unclicked URL w.r.t to the entire URL's available. And finally we have analyzed our results by comparing two factors *m* and *n* which are nothing but distance between clicked URL and original results and distance between clicked URL and our refined results.

Entire work we have done is offline we can make it online by integrating this work to some work search engine. We have refined the results by considering only title and snippet to entire document of the URL and its internal links also.

**Acknowledgment**

I acknowledge **Mr. SABHAVAT GOPAL NAIK,** a final year post graduate, for giving me confidence in myself and helped me see what I could be. I would like to mention his help in collecting "CILCK THROUGH DATA" which is required for my work.

## REFERENCES

[1] Zheng Lu,Hongyuan Zha,Xiaokang Yang,Weiyao Lin, "A New Algorithm for Inferring User Search Goals with Feedback Sessions", IEEE transactions on knowledge and data engineering VOL.25,NO.3 (March 2013)
[2] U. Lee, Z. Liu, and J. Cho, "Automatic Identification of User Goals in Web Search," Proc. 14th Int'l Conf. World Wide Web (WWW '05), pp. 391-400, 2005.
[3] www.google.com
[4] Information storage and retrieval systems Theory and implementation 2nd edition by gerald J.kowalski and mark T.maybury.
[5] Gurbunder Kaur,"similarity measures of different types of fuzzy sets",thesis report of master of science.
[6] http://www.seoquake.com/