IPHV7I10029X
International Journal Of Advanced Research and Innovation -Vol.7, Issue .I
ISSN Online: 2319 – 9253
Print: 2319 – 9245

# IDENTIFICATION OF USER SEARCH GOALS IN WEB SEARCH ENGINE

[#1]G.Sruthikala-M.Tech Student,
[#2]G.Prasad-Assoc. Professor,
[#3]M.Sreelatha-Assoc.Professor, HOD,
Department of CSE,
SAHAJA COLLEGE OF ENGINEERING &TECHNOLOGY FOR WOMEN,
KARIMNAGAR TELANAGANA.

***Abstract*— Nowadays Internet is widely used by users to satisfy various information needs. However, ambiguous query/topic submitted to search engine doesn't satisfy user information needs, because different users may have different information needs on diverse aspects upon submission of same query/topic to search engine. So discovering different user search goals becomes complicated. The evaluation and depiction of user search goals can be very useful in improving search engine relevance and user knowledge. This paper proposes a novel approach for inferring user search goals by analyzing user query logs from various search engines. The proposed approach is used to discover different user search goals for a query by clustering the user feedback sessions. Feedback sessions are constructed from clickthrough logs of various search engines. The method first generates pseudo-documents to better represent feedback sessions for clustering. Finally, clustering pseudo-documents to discover different user search goals and depict them with some keywords. Then these user search goals are used to restructure the web search results.**

***Keywords*— User search goals, implicit feedback sessions, pseudo-documents, restructuring search results, k-means clustering.**

## I. INTRODUCTION

In web based search applications, user submits the query to search engine to search efficient information. The information needs of different user may differ in various aspects of query information. This becomes difficult to achieve user information needs. Sometimes ambiguous queries may not exactly represented by users so it results in less understandable to search engine. To achieve the user specific information needs many ambiguous/uncertain queries may cover a broad topic and dissimilar users may want to get information on different aspects when they submit the same query. For example, when user submits a query "java" to search engine, some users are interested to know information about programming language and some users want to know information about island of Indonesia. Therefore, it is necessary to discover different user information search goals. User information need is to desire and obtain the information to satisfy the needs of each user. To satisfy the user information needs by considering the search goals with user given query, cluster the user information needs with different search goals. Because the interference and evaluation of user search goals with query might have a numeral of advantages in improving the search engine significance and user knowledge. So it is necessary to collect the different user goal and retrieve the efficient information on different aspects of a query. Capturing different user search goals related to information needs changes the normal query based information retrieval.

Evaluation and analysis of user search goals has many advantages as follows.

- Reorganize web search results according to user search goals by grouping search results with same information need. This can be useful to other users with different search goals to find easily what they want.
- Query recommendation by using user search goals depicted with some keywords. This can be helpful to other users to form their query more effective.
- Reranking web search results according to different user search goals.

User search goal analysis is important to optimize search engine and effective query results organization. When query is submitted to search engine, the returned web pages of search results are analyzed [3], [4]. Since it does not consider user feedback, many unuseful and noisy search results that are not clicked by user may be analyzed. This may degrade the search goals discovery. X. Wang and C-X. Zhai [2] learns interesting aspects of similar query/topic from web search logs which consists clicked web pages URLs and organize search results accordingly. Their approach may results in limitation, as the different clicked URLs for a query/topic may be small in number. There are many works [11], [12] which classify queries into some predefined specific classes and try to find out query intents and user goals. However, different queries have different search goals and finding precise, suitable predefined search goal classes may be difficult and sometimes impossible to categorize.

Clustering search results is an efficient method to systematize search results, which allows a user to find the way into applicable documents quickly. In this paper, our aim is to discover different user search goals for a query and depict each search goal with some keywords automatically. To discover the user information automatically at different point of view with user given query and collects the similar search goal result with URL first we collect similar feedbacks sessions from user click-through logs of different search engines. Then, map feedback sessions to pseudo-documents which reflects user information needs. At last, k means clustering algorithm can be used to cluster these pseudo-documents for inferring user search goals and depicting them with some meaningful keywords. Then these search goals can be used to restructure the web search results.

The rest of the paper is organized as follows: Section II contains literature survey about related work. Section III contains description of the proposed system. Finally paper is concluded in the Section IV.

## II. Litrature Survey

Since many years, research in web log mining has been subject of interest. Many previous works has been investigated on problem of analyzing user query logs [5], [9], [10], [12], [13]. The information in query logs has been used in many different ways, such as to infer search query intents or user goals, to classify queries, to provide context during search, to facilitate personalization, to suggest query substitutes and to identify frequently asked questions (FAQs).

Effective organization of search results is critical for improving utility and relevance of any search engine. Clustering search results is an effective way to organize search results which allows a user to navigate into relevant documents quickly. Generally all existing work [3], [17] perform clustering on a set of top ranked results to partition results into general clusters, which may contain different subtopics of the general query term. However, this clustering strategy has two deficiencies which make it not always work well. First, discovered clusters do not necessarily correspond to the interesting aspect of a topic from user-oriented perspective. Second, cluster labels are more general and not informative to identify appropriate clusters. Wang and Zhai [2] proposed approach to organize search results in user-oriented manner. They used search engines log to learn interesting aspects of similar queries and categorize search results into aspects learned. Cluster labels are generated from past query words entered by users.

H-J Zeng et.al [3] proposed a query based method to cluster search results. For a given query, the rank list of documents return by a certain Web search engine, it first extracts and ranks most salient phrases as candidate cluster names, base on a regression model learned from pervious training data. Candidate clusters are formed by assigning documents to relevant salient phrases and the final cluster are generated by merging these candidate clusters. But this method only produces the result with higher level of the documents only and it doesn’t make the results for all search based user goals.

H. Chen and S. Dumais [4] developed a user interface that organizes web search results into hierarchical categories. Automatic text classification technique (SVM classifier) was used to classify arbitrary search results into existing category structure on-the-fly. This approach has advantage of known category labels information, for classifying new items into the category structure and to help user to quickly focus on task relevant information. A user study compared new category interface with the traditional ranked list interface of search results, which showed that category interface is superior in both subjective and objective manner.

T. Joachims [5] proposed an approach to automatically optimizing the retrieval quality of search engine using click-through data stored in query logs and the log of links the users clicked on in presented ranking. Taking support vector machine (SVM) approach, for learning ranking functions in information retrieval.

T. Joachims et al. [6] did a lot of work on examining the reliability of implicit feedback generated from clickthrough data in www search. The author proposes strategy to automatically generate training examples for learning retrieval functions from observed user behavior. The user study is intended to examine how users interrelate with the list of ranked results from the Google search engine and how their behavior can be interpreted as significance judgments. Implicit feedback can be used for evaluating quality of retrieval functions [7].

Preceding studies encompass mainly focused on manual query-log investigation to recognize Web query goals. U. Lee et al. [11] studied the “goal" at the back based on a user's Web query, so that this goal can be used to get better the excellence of a search engine's results. Their proposed method identifies the user goal automatically with no any explicit feedback from the user.

User may issue number of queries to search engine in order to achieve information need/tasks at a variety of granularities. R. Jones and K.L. Klinkner [15] proposed a method to detect search goal and mission boundaries for automatic segmenting query logs into hierarchical structure. Their method identifies whether a pair of queries belongs to the same goal or mission and does not consider search goal in detail.

Zamir et al. [17] used Suffix Tree Clustering (STC) to identify set of documents having common phrases and then create cluster based on these phrases or contents. They used documents snippets instead whole document for clustering web documents. However, generating meaningful labels for clusters is most challenging in document clustering. So, to overcome this difficulty, in [3], a supervised learning method is used to extract possible phrases from search result snippets or contents and these phrases are then used to cluster web search results.

## III. Proposed System

In this section, basic operations involved in proposed approach to discover user search goals/intents by clustering pseudo-documents are described. The flow of the proposed system design will be as shown in Fig. 1.

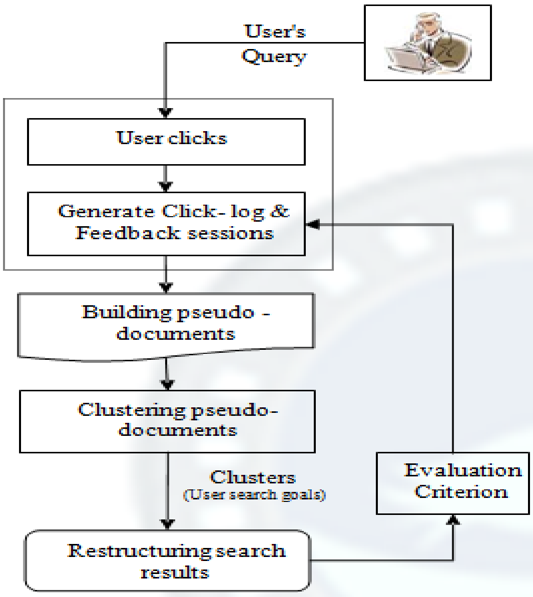
User's Query
User clicks
Generate Click- log & Feedback sessions
Building pseudo - documents
Clustering pseudo- documents
Clusters (User search goals)
Evaluation Criterion
Restructuring search results

Fig. 1 Flow Diagram of Proposed System

### A. Click through data

In web search environment, there are many abundant queries and user clicks. User clicks represent implicit relevance feedback. In this framework, user clicks are recorded in user clickthrough data. User uses clickthrough data stored in user logs to simulate user experience in web search. In general, when query is issued, the user usually scans links to documents in a result list from first to last. Clearly, the user clicks on the links to the documents that look relevant of informed choice and skips other documents. Therefore, the proposed approach utilize user click as relevance judgments to evaluate search precision since clickthrough data can be collected at low cost, it is possible to do large scale evaluation under this framework.

*1).Feedback sessions:* Feedback sessions are considered as users' implicit feedback. In general, a session for web search is a sequence of consecutive queries to satisfy single information and some clicked results. But to infer user search intents/goals for a particular query, single session is considered. Single session corresponds to only one query, which differs from conservative session. The proposed feedback session consists of both clicked and unclicked URLs for a particular query in a single session and ends with last clicked URL. This shows that before last clicked URL, all the URLs are scanned and evaluated by user. Therefore, all clicked URLs and unclicked URLs before last click are considered as user feedbacks. In each feedback session clicked URL (visited link) tells users information need and unclicked URL (unvisited link) tells what users do not want. This visited link is called as positive feedback and unvisited link is called as negative feedback. There are large numbers of diverse feedback sessions in user clickthrough log. So it is efficient to examine feedback sessions for inferring user search goals than to examine clicked URLs or search results directly.

### B. Building pseudo-documents

As URLs alone are not informative enough to tell intended meaning of a submitted query. To obtain rich information, we enrich each URL with additional text content by extracting the titles and snippets of URLs appearing in feedback session. Thus, each URL in feedback session is represented by small textual content which contains its title and snippet. Then some text preprocessing is done on those textual contents, such as transforming all letters to lowercase, eliminating stop words (frequent words) and word stemming by using porter algorithm [16]. Lastly, TF-IDF [1] vector of URL's titles and snippets are formed respectively as,

$$T_{u_i} = [t_{w_1}, t_{w_2}, \dots\dots, t_{w_n}]^T \quad (1)$$

$$S_{u_i} = [s_{w_1}, s_{w_2}, \dots\dots, s_{w_n}]^T$$

where $T_{u_i}$ and $S_{u_i}$ are TF-IDF vectors of URL's title and snippet, respectively. $u_i$ is $i^{th}$ URL in feedback session. $W_j$ is the $j^{th}$ term in the enriched URL. The $t_{w_j}$ and $s_{w_j}$ denotes $j^{th}$ term in the URL's title and snippet respectively. Feature representation $F_{u_i}$ of $i^{th}$ enriched URL is weighted sum of $T_{u_i}$ and $S_{u_i}$.

$$F_{u_i} = w_t T_{u_i} + w_s S_{u_i} = [f_{w_1}, f_{w_2}, \dots\dots, f_{w_n}]^T \quad (2)$$

where $w_t$ and $w_s$ are weights of title and snippet respectively. Each term of $F_{u_i}$, denotes importance of term in $i^{th}$ URL.

In order to obtain feature representation of a feedback session, optimization method is used to merge feature representations of each clicked and unclicked enriched URLs in the feedback session. Let $F_{fs}$ be feature representation of a feedback session, $F_{uc_i}$ and $F_{u\bar{c}_j}$ are feature representation of clicked and unclicked URLs respectively and $f_{fs}(w)$ is value for term w. $F_{fs}$ should be such that sum of distance between $F_{fs}$ and each $F_{uc_i}$ is minimized and sum of distance between $F_{fs}$ and $F_{u\bar{c}_j}$ is maximized.

$$F_{fs} = [f_{fs}(w_1), f_{fs}(w_2), \dots\dots, f_{fs}(w_n)]^T \quad (3)$$

Each feedback session is represented by $F_{fs}$. This is nothing but pseudo-document which is used for discovering user intents or search goals. These pseudo-documents contain what user requires and what do not, which is used to learn interesting aspects of a query.

### C. Clustering pseudo-documents with K-means

In order to cluster pseudo-documents with k-means, the important factor is to define the distance measure between two data points and defining the number of clusters. Thereare two variations of distance measures, one is derived from cosine based similarity and the other is derived from Jacquard I similarity coefficient. The feature representation of pseudo-document is The similarity between two pseudo-documents is defined as below:

$$\mathbf{F}_{fs}$$

$$Sim_{i,j} = \cos\left(\mathbf{F}_{fs_i}, \mathbf{F}_{fs_j}\right) = \frac{\mathbf{F}_{fs_i} \cdot \mathbf{F}_{fs_j}}{|\mathbf{F}_{fs_i}||\mathbf{F}_{fs_j}|}$$

OR,

$$Sim_{i,j} = \mathrm{jac}\left(\mathbf{F}_{fs_i}, \mathbf{F}_{fs_j}\right) = \frac{|\mathbf{F}_{fs_i} \cap \mathbf{F}_{fs_j}|}{|\mathbf{F}_{fs_i} \cup \mathbf{F}_{fs_j}|} \tag{4}$$

And the distance two feedback sessions i.e. pseudo-documents is

$$Dist_{i,j} = 1 - Sim_{i,j} \tag{5}$$

K-means algorithm is used to cluster pseudo-documents because of its simplicity and effectiveness. K-means clustering results in good quality performance for document clustering. As a prior number of user search goals for a query are unknown so we have chosen arbitrary value for k initially (i.e. 1, 2, 3, 4, 5). Then, perform clustering on these five different values. The optimal value for k is determined by evaluation criterion.

After clustering all pseudo-documents, each cluster denotes user search goal i.e. intention of user. Centroid of a cluster is calculated by taking average of all the vectors of the pseudo- documents in the cluster,

$$\mathbf{F}_{center_i} = \frac{\sum_{k=1}^{C_i} \mathbf{F}_{fs_k}}{C_i}, (\mathbf{F}_{fs_k} \subset Cluster\ i) \tag{6}$$

where $\mathbf{F}_{center_i}$ is i<sup>th</sup> cluster center and $C_i$ is the number of pseudo-documents in the i<sup>th</sup> cluster. $\mathbf{F}_{center_i}$ is used represent user search goal/intent of i<sup>th</sup> cluster and to categorize the search results. User search goals/intents depicted with the terms with highest values in the center points of each cluster. These depicted keywords can be used to suggest more meaningful and precise query.

### *D. Restructuring web search results*

Web search results are reorganized on the basis of discovered user search goals/intents. As inferred user search goals are depicted with vectors in (6) and feature representation of each URL in search result is calculated by (1) and (2). Then categorize each URL into a cluster centered with user search goals/intents by selecting smallest distance between user search goal vectors and URL vectors.

### *E. Evaluation criterion*

The performance of restructured (clustered) web search results and original search results is evaluated by using parameters like Average Precision (AP) [1], Voted AP (VAP) which is AP of the class having more clicks, Risk to avoid wrong classification of search results and Classified AP (CAP). If user got correct classified results with higher CAP value, this value is used to optimize the no of clusters of user search goals.

1 ) *Average precision (AP):* It is calculated according to given user feedbacks. AP is the average of precisions computed at the point of each clicked document in the ranked sequence of user feedback.

$$AP = \frac{1}{N^+} \sum_{r=1}^{N} rel(r) \frac{R_r}{r}$$

where $N^+$ is the number of clicked documents from total retrieved documents in single user feedback session, *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 *Rr* is the number of relevant retrieved documents of rank *r* or less.

2) *Voted AP (VAP):* It is calculated for restructured search results classes i.e. different clustered results classes. It is same as AP and calculated for class which having more clicks i.e. the class user interested in.

$$VAP = \frac{1}{NC} \sum_{r=1}^{L} rel(r) \frac{R_r}{r}$$

$$VAP = \frac{1}{NC} \sum_{r=1}^{L} rel(r) \frac{R_r}{r}$$

where NC is the number of clicked documents from the class having maximum number of clicks.

3) *Risk:* Sometimes VAP will always be highest value because each URL from single session is classified into the single class no matter whether users have different search goals or not. So, there should be a risk to avoid wrong classification search results into too many classes. It evaluates the normalized number of clicked URL pairs that are not in the same class.

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

where m is number of clicked URLs and $d_{ij}$ is 0 if pair of clicked URLs belongs to same class otherwise $d_{ij}$ is 1.

4) *Classified AP (CAP):* New criterion Classified AP (CAP) is extension of VAP by using above Risk. It combines AP of class having more clicks and risk of wrong classification. It is used to evaluate performance of restructured search results.

$$CAP = VAP \times (1 \quad Risk)^{\gamma}$$

where γ is normalizing factor used to adjust influence of Risk on CAP.

## IV. Conclusions

The proposed system can be used to improve discovery of user search goals for a query by clustering user feedback sessions represented by pseudo-documents. Using proposed system, the inferred user search goals/intents can be used to restructure web search results. So, users can find exact information needed as they want very efficiently. The discovered clusters can also be used to assist users in web search.

# REFERENCES

[1] R. Baeza-Yates and B. Ribeiro-Neto, *Modern Information Retrieval*.ACM Press, 1999.

[2] 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.

[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] 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.

[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] T. Joachims, *"Evaluating Retrieval Performance Using Click-through Data"*, Text Mining, J. Franke, G. Nakhaeizadeh, and I. Renz, eds., pp. 79-96, Physica/Springer Verlag, 2003.

[8] Zheng Lu, Hongyuan Zha, Xiaokang Yang, Weiyao Lin, 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.

[9] D. Beeferman and A. Berger, "*Agglomerative Clustering of a Search Engine Query Log,*" Proc. Sixth ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining (SIGKDD '00), pp. 407-416, 2000.

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

[11] 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.

[12] D. Shen, J. Sun, Q. Yang, and Z. Chen, "*Building Bridges for Web Query Classification*," Proc. 29th Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR '06), pp. 131-138, 2006.

[13] 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.

[14] C.-K Huang, L.-F Chien, and Y.-J Oyang, "*Relevant Term Suggestion in Interactive Web Search Based on Contextual Information in Query Session Logs,*" J. Am. Soc. for Information Science and Technology, vol. 54, no. 7, pp. 638-649, 2003.

[15] 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.

[16] Porter, M. *An algorithm for suffix stripping*. Program, Vol. 14(3), pp. 130-137, 1980.

[17] O. Zamir and O. Etzioni. *Grouper: A dynamic clustering interface to web search results.* Computer Networks, 31(11-16), pp.1361-1374, 1999.