Extracting User Search Goals by Conceptually Related Feedback Sessions

Kameswari Sriramagiri
M.tech in Software Engineering
Aurora's Technological & Research Institute,
parvathapur, uppal, Hyderabad-500039

N.Nirmala Jyothi
ASSISTANT PROFESSOR in CSE DEPARTMENT
Aurora's Technological & Research Institute,
parvathapur, uppal, Hyderabad-500039

Abstract: This research proposal proposes to enhance search query log analysis by taking into account the conceptual properties of query terms. We first describe a method for extracting a conceptual representation of a search query log and then show how we can use it to relatively extract results with no ambiguity. The concept relation is composed of a set terms frequent together concept relations and of a function to measure the concept distance between terms, which further referred as CTF (concept term frequency). We then sink CTF with feedback sessions, which we build upon click through logs. Further a query terms clustering algorithm that is applied to the log representation to extract user interests.

Keywords— User search goals, implicit feedback sessions, pseudo-documents, restructuring search results,

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.

II. Related Work:

Mining processes can be applied to large search query logs in order to extract knowledge about user interests [1], [2], [3], [4], [5], [6], [7], [8], [9] and [10]. This is in particular a necessary step for the design of true user centric applications in which user search behaviors are identified and taken into account. In recent years, much research has been done in the domain of search query logs analysis. To date, researchers have mostly focused on statistical methods for extracting knowledge from these data. These proposals are not applicable to problems related to the semantics of the data such as the identification of users search interests.

In order to handle this issue, Zheng Lu et al [11], proposed A New Algorithm for Inferring User Search Goals with Feedback Sessions. First, this model introduces feedback sessions to be analyzed to infer user search goals rather than search results or clicked URLs. Both the clicked URLs and the un-clicked ones before the last click are considered as user implicit feedbacks and taken into account to construct feedback sessions. Therefore, feedback sessions can reflect user information needs more efficiently. Second, it maps feedback sessions to pseudo documents to approximate goal texts in user minds. The pseudo-documents can enrich the URLs with additional textual contents including the titles and snippets. Based on these pseudo-documents, user search goals can then be discovered and depicted with some keywords. Finally, a new criterion CAP is formulated to evaluate the performance of user search goal inference.

III. Extracting User Search Goals by Conceptually Related Feedback Sessions

With the motivation gained from "A New Algorithm for Inferring User Search Goals with Feedback Sessions" [11], Here we devised a conceptually related feedback sessions approach to extract unambiguous user search goals. Feedback sessions that devised in research article considered as motivation will refined such that the user clicks considered to build feedback sessions should be conceptually related. This we achieve by using a metric called concept term frequency. The proposal is aimed to achieve (i) the metric the considered in the proposal helps to improve the conceptual relevancy of the clicks collected under one particular feedback session and (ii) to limit the size of the feedback session, to stabilize the performance and scalability of the proposed model.

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.

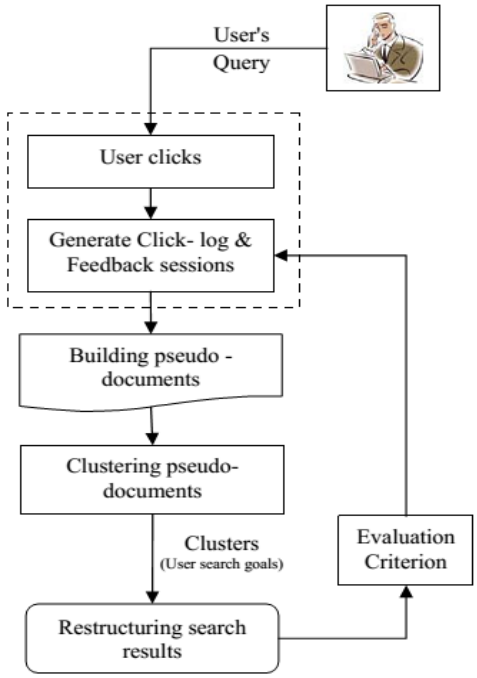

Fig. 1 Flow Diagram of Proposed System

Clickthrough 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.

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.

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. Lastly, TF-IDF [1] vector of URL's titles and snippets are formed respectively as,

$$T_{u_i} = [T_{w_1}, T_{w_2}, ..., T_{w_n}]^T$$

$$S_{u_i} = [S_{w_1}, S_{w_2}, ..., S_{w_n}]^T$$

Where T_{u_i} and S_{u_i} are TF-IDF vectors of URL's title and snippet, respectively. u_i Is a i^h URL in feedback session. W_j is the j^h term in the enriched URL. The T_{w_n} and S_{w_n} denotes j^h term in the URL's title and snippet respectively. Feature representation, of i^h enriched URL is weighted sum of T_{u_i} and S_{u_i} .

IV. Predicting Degree of context sensitivity between feedback sessions and search key phrases

The strategy of computing context similarity score (CSS) endorsed here. Right here with regards to CSS we contemplate the bipartite graph to signify the context similarity weights.

Assumptions:

Let set of user sessions $us_1, us_2, us_3, ...us_n$

Let set of search query phrases $kp_1, kp_2, kp_3, ..., kp_n$, such that these search query phrases belongs either one or more of the search queries

Process

In the process of detecting the closeness of each search query phrase with user sessions, initially we build a bipartite weighted graph between user sessions and the search query phrases. The number of context similarities required for influenced user sessions for each search query phrase is considered to be as edge weight that connects the related search query phrase and user session.

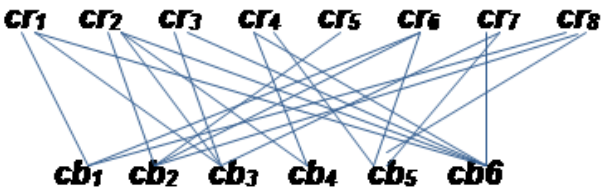

Fig 2: bipartite graph between user sessions and search query phrases

If a search query phrase kp_1 influenced to revise a user session us_1 then the weight of the connection between kp_1 and us_1 will be the no of context similarities was made to that user session us_1 due to the search query phrase kp_1 , the context similarities r will be adjusted to threshold rt ($0 \le rt < 1$) (see Eq1).

$$rt = 1 - \frac{1}{r} \ldots \text{(Eq1).}$$

Let matrix A representing the connection weights (Matrix A) between each search query phrase cr and each user session cb

Let matrix A' be the transpose of matrix A that signifying the connection between a user sessions and each search query

Let consider a set of user sessions US as a database and depict it as a bipartite graph without loss of information. Let $US = \{us_1, us_2, us_3, ..., us_m\}$ be a list of influenced user sessions and $KP = \{kp_1, kp_2, kp_3, ..., kp_n\}$ be the corresponding search query phrases. Then, clearly CB is equivalent to the bipartite weighted graph $G = (US, KP, E)$ where

$$E = \{(us, kp) : ew(kp, us) > 0, us \in US, kp \in KP\}$$

.

Here $ew(kp, us)$ is weight of the edge between search query phrase kp and user session us

The graph representation (see fig 2) indicates the bipartite relation between search query phrases and user sessions. Context similarity weights of the different user sessions represent their importance. Intuitively, a user session with high context similarity weight is affected to multiple context similarities due to search query phrases with high context similarity score. The underpinning association of user sessions and search query phrases is that of association between hubs and authorities [12].

The formulated strategy of distinguishing user sessions context similarity weights using bipartite graph is explored below:

The matrix structure of weight of the edge amongst user sessions and search query phrases in bipartite graph. The edge weight indicates the no of context similarities occurred to that user session due to the connected search query phrase.

Each hub (user session) weight primarily regarded as 1

As introduced in [12] criteria, find Authority (feature) weights by matrix multiplication of A' and hw. The consequent matrix aw is authority weights. And then exact hub weights tends to be found by multiplying matrix A with matrix aw

$$hw = A \times aw$$

Then the context similarity score css of search query phrase kp can be determined as follows

$$css(kp) = \frac{\sum_{i=1}^{m} \{hw(us_i) : (ew(kp, us_i) > 0)\}}{\sum_{i=1}^{m} hw(us_i)}$$

Here in the above equation, $ew(kp_j, us_i)$ is weight of the edge between search query phrase kp and user session us_i

Finding Degree of context sensitivity of User sessions

Then Degree of context sensitivity dcs of each user session can be found as follows:

$$dcs(us_i) = 1 - \frac{\sum_{j=1}^{m} \{css(kp_j) : (ew(kp_j, us_i) > 0\}}{|KP|}$$

Here in the above equation $|KP|$ indicates the total number of search query phrases considered.

Here in the above equation $ew(kp_j, us_i)$ is weight of the edge between search query phrase kp_j and user session us_i

Then the degree of context sensitivity threshold of user sessions can be found as follows:

$$dcst_{us} = \frac{\sum_{i=1}^{|CB|} dcs(us_i)}{|US|}$$

Right here in the preceding formula $|US|$ signifies the total quantity of user sessions

The degree of context sensitivity range of user sessions can be explored as follows

Lower threshold of $dcst_{us}$ range is

$$dcst_l(us) = dcst_{us} - \left(\frac{\sum_{i=1}^{|US|} dv(us_i)}{|US|} \right)$$

Higher threshold of $dcst_{us}$ range is

$$dcst_h(us) = dcst + \left(\frac{\sum_{i=1}^{|US|} dv(us_i)}{|US|} \right)$$

User session us can be said as dissimilar if and only if $dcs(us) < dcst_l$

User session us can be said as highly relevant if and only if

$$dcs(us) \geq dcst_l(us) \,\&\&\, dcs(us) < dcst_h(us)$$

User session us can be confirmed as possibly relevant if $dcs(us) \geq dcst_h(us)$

Finding Degree of context sensitivity of Search query phrases

In this similar passion degree of context sensitivity of search query phrases can also be measured. The exploration of finding degree of context sensitivity of search query phrases is follows

Degree of context sensitivity dcs of each user session can be found as follows:

$$dcs(kp_i) = 1 - \frac{\sum_{j=1}^{m} \{dcs(us_j) : (ew(kp_i, us_j) > 0\}}{|US|}$$

Right here in the preceding formula $|US|$ signifies the overall total of user sessions considered.

Right here in the preceding formula $ew(kp_i, us_j)$ is weight of the edge between search query phrase kp_i and user session us_j

Then the degree of context sensitivity threshold of search query phrases can be found as follows:

$$dcst_{kp} = \frac{\sum_{i=1}^{|KP|} dcs(kp_i)}{|KP|}$$

Right here in the preceding formula $|KP|$ signifies the overall total of search query phrases

The $dcst_{kp}$ value discovered out of our illustrative example is 0.880644783

The degree of context sensitivity range of search query phrases can be explored as follows

Lower threshold of $dcst_{kp}$ range is

$$dcst_l(kp) = dcst_{kp} - \left(\frac{\sum_{i=1}^{|KP|} dv(kp_i)}{|KP|} \right)$$

Higher threshold of $dcst_{kp}$ range is

$$dcst_h(kp) = dcst_{kp} + \left(\frac{\sum_{i=1}^{|KP|} dv(kp_i)}{|KP|} \right)$$

Finding Degree of context sensitivity of Search queries

After finding the degree of context sensitivity of search query phrases, the degree of context sensitivity of search queries can be found as follow:

Let $kp_1, kp_2, kp_3, ...kp_n$ be the search query phrases of search query sq_i , then the degree of context sensitivity of the search query sq_i can be found as follows:

$$dcs(sq_i) = \frac{\sum_{j=1}^{|sq_i|} \{dcs(kp_j) : kp_j \in sq_i\}}{|sq_i|}$$

Here in above Equation $dcs(sq_i)$ is the degree of context sensitivity of search query sq_i , $dcs(kp_j)$ is the degree of context sensitivity of search query phrase kp_j , which comes under search query sq_i . The total number of search query phrases found under search query sq_i represented by sq_i .

Search query sq_i can be said as non relevant if and only if $dcs(sq_i) \le dcst_l(kp)$

Search query sq_i can be said as possible to possibly relevant if and only if

$$dcs(sq_i) \ge dcst_l(kp) \,\&\&\, dcs(sq_i) < dcst_h(kp)$$

Search query sq_i can be confirmed as highly related if $dcs(sq_i) \ge dcst_h(kp)$

V. Exploration of Experimental Results:

The feedback sessions that considered through click-through log has been used for evaluation. The experiments were conducted on divergent number of results acquired against search queries. The explored results indicating that the refinement of the feedback sessions lead to improve the search results cluster accuracy. The metrics that we consider to claim the impact of evaluating context sensitivity of the feedback sessions are cluster correctness and correlation factor of clusters (see figure 3 and 4).

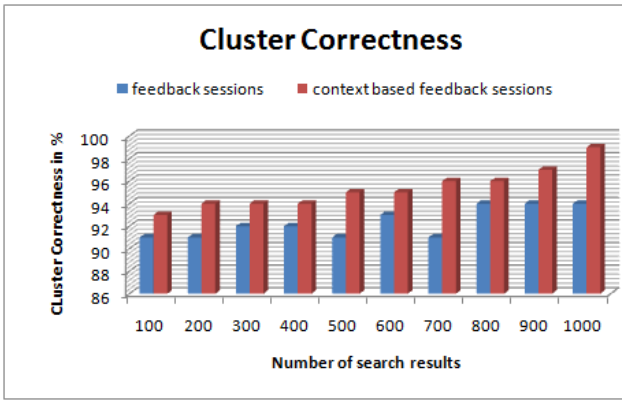

Fig 3: The representation of performance advantage towards cluster correctness

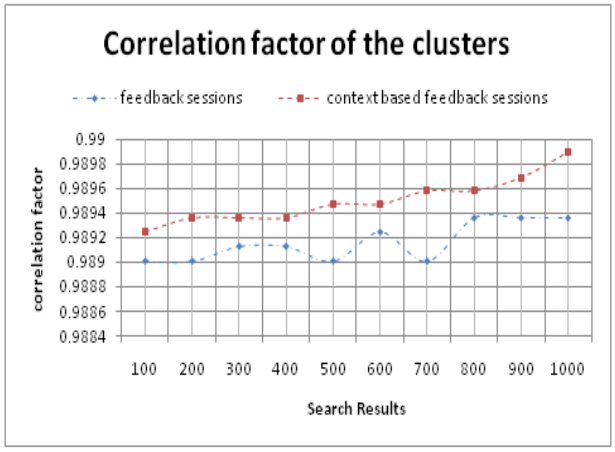

Fig 4: The correlation factor of the clusters formed

VI. Conclusion

Here in this paper we proposed a novel statistical model to identify the user search goals by feedback sessions, which is an improvised version of the model devised in [11]. In our model, rather using all feedback sessions, we refined these by verifying context sensitivity between search queries and feedback sessions. The devised model is the motivation of the procedure explored in [12]. The experimental results indicating that evaluating the context sensitivity led to improve the cluster correctness and cluster correlation factor.

References:

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

[2]. 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.

[3]. 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.

[4]. 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.

[5]. 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.

[6]. 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.

[7]. X. Li, Y.-Y Wang, and A. Acero, "Learning Query Intent from Regularized Click Graphs,"Proc. 31st Ann. Int'l ACM SIGIR Conf. Research and Development in Information Retrieval (SIGIR '08), pp. 339-346, 2008.

[8]. M. Pasca and B.-V Durme, "What You Seek Is what You Get: Extraction of Class Attributes from Query Logs,"Proc. 20th Int'l Joint Conf. Artificial Intelligence (IJCAI '07),pp. 2832-2837, 2007.

[9]. B. Poblete and B.-Y Ricardo, "Query-Sets: Using Implicit Feedback and Query Patterns to Organize Web Documents,"Proc. 17th Int'l Conf. World Wide Web (WWW '08),pp. 41-50, 2008.

[10]. 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.

[11]. Zheng Lu; Hongyuan Zha; Xiaokang Yang; Weiyao Lin; Zhaohui Zheng, "A New Algorithm for Inferring User Search Goals with Feedback Sessions," Knowledge and Data Engineering, IEEE Transactions on ,

vol.25, no.3, pp.502,513, March 2013 doi: 10.1109/TKDE.2011.248

[12]. Rudra Kumar M, Ananda Rao A; Assessing the Fault Proneness Degree (DFP) of Change Requests and Change Request Artifacts: A statistical bipartite graph strategy;Eighth International Conference on Data Mining and Warehousing (ICDMW 2014).

