A Novel Method for Inferring User Query Substitutions with Feedback Terms

P. N. Santosh Kumar[1], Assistant Professor, Dept. Of ECM, SNIST, Hyderabad, India
Kashif Ali[2], **Akash Sharma**[3], Final year UG students, Dept. of ECM, SNIST, Hyderabad, India

Abstract

During the submission of a topic or an ambiguous query to a search engine, the users may have many search goals. The conclusion and analysis of users search goals can be very useful in improving the performance of a search engine. To conclude these goals by analyzing search engine query logs, a novel approach is proposed. In this paper, a framework is proposed to find various search goals for a query by clustering the feed-back sessions. These Feedback sessions are built from user clicks-through data and can efficiently reflect the required information. Secondly, a novel approach is proposed to generate pseudo-documents by using feedback sessions for clustering. For clustering a novel algorithm, i.e. bisecting K-means algorithm is used. At the end, a new criterion “Classified Average Precision (CAP)” is proposed to evaluate the performance of inferring user search goals.

Keywords: User search goals, Pseudo documents, Feedback sessions, Classified Average Precision.

I. Introduction

This approach is used to infer user search goals for a query by clustering proposed feedback sessions. Feedback session is series of clicked and un-clicked URLs’. Clustering algorithm is applied on to pseudo-documents which are generated from feedback session. It forms clusters according to the user search goals or queries. Finally evaluation criterion is used to check the performance of the system. It compares the performance of k-means and bisecting k-means algorithm. Bisecting k-means gives the better performance than k-means. The objectives of this project are:

1. Feedback Session: The proposed feedback session consists of both clicked and un-clicked URLs and ends with the URL that was clicked in a single session at last. It is motivated that before the last click, all the URLs have been scanned and evaluated by the users. Like this, it shows the list of clicked and un-clicked URL’s by the user.

2. Optimization method to map: Feedback session is mapped onto pseudo document, which consists of titles and snippets by using optimization method. Bisecting K-Means Clustering Algorithm is applied to cluster pseudo document.

3. Evaluation Criterion

To check performance of system an evaluation criterion is used

II. Literature Survey

Many works regarding user search goals analysis have been investigated. This can be summarized into two classes: classification of query and search result reorganization.

(i) Topic wise classification of web queries has drawn recent interest because of the promise it offers in improving retrieval effectiveness and efficiency. However, much of this promise depends on whether classification is performed before or after the query is used to retrieve documents.

(ii) Effective organization of search results is critical for improving the utility of any search engine. Clustering of search results is an effective way to organize search results that allows a user to navigate into relevant documents quickly. However, there are two deficiencies of this approach: (a) the clusters discovered do not necessarily correspond to the interesting aspects of a topic from the user's perspective; and (b) the cluster labels generated are not informative enough to allow user to identify the right cluster. In this paper, we propose to address these two deficiencies by learning interesting aspects of a topic from Web search logs and organizing search results accordingly; and generating more meaningful cluster labels using past query words entered by users. We try to evaluate our proposed method on a commercial search engine data. Compared with the traditional methods of clustering of search results, our method can obtain better result organization and more meaningful labels. People try to reorganize search results, but this involves many noisy search results that are not clicked by any user. In the third class, people aim at detecting session boundaries. However, this only identifies whether pair of queries belongs to the same goal and does not care what the goal is in detail. To improve the performance of the system we implemented a new bisecting K-means clustering algorithm

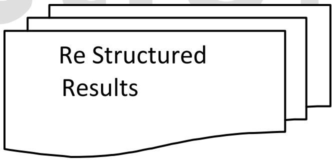
PREVIOUS RESEARCH PAPERS	RESULT/CONCLUSION
Z. Chen	Worked on Query classification Limitations- Experiment was conducted on a potentially-biased dataset
H. Chen	Organizes search results into a hierarchical category structure. Limitations- Query aspects without user feedback have limitations to improve search engine relevance
Wang , Zhai	clustered queries and learned aspects of similar queries Limitations- This method does not work if we try to discover user search goals of any one single query in the query cluster rather than a cluster of similar queries
R. Jones and K.L. Klinkner	Introduce search goals and missions to detect session boundary hierarchically Limitations- Their method only identifies whether a pair of queries belong to the same goal or not and does not care what the goal is in detail

Table1. Literature Survey

III. PROPOSED WORK

Initially, the user logs into the system and searches for the required documents by submitting an ambiguous query. From the system generated results user clicks some desired URLs, by using this clicked data system creates a feedback session. Once the feedback session is mapped to pseudo document, then clustering is performed. At last, performance of the system is calculated by using CAP evaluation criteria.

Here, we describe the proposed feedback sessions and then introduce the proposed -documents to represent feedback sessions.



III.I FEEDBACK SESSION

The proposed feedback session consists of both clicked and un-clicked URLs and ends with the last URL that was clicked in a single session. It is motivated that before the last click of URL, all the URLs have been scanned and evaluated by the users. Therefore, besides the clicked URLs, the un-clicked ones before the last click should be a part of the user feedbacks. Fig. 5.1 shows an example of a feedback session and a single session.

In Fig.1, the left part lists 10 search results of the query “the sun” and the right part is a user’s click sequence where “0” means “un-clicked.” The single session includes all the 10 URLs in Figure.2, while the feedback session only includes the seven URLs in the upper rectangular box. Out of seven URLs, three are clicked URLs and four are un-clicked URLs in this example. Since users will scan the URLs one by one from top to bottom, we can assume that besides the three clicked URLs, the four un-clicked URLs in the rectangular box have also been browsed and evaluated by the user and they should be a part of the user feedback. In the part of feedback session, the clicked URLs tell what users require and the un-clicked URLs reflect what users do not care about. It should be noted that the un-clicked URLs after the last clicked URL should not be included into the feedback sessions since it is not certain whether they were scanned or not. Each feedback session can tell what a user requires and what he/she does not care about. Lots of diverse feedbacks sessions in user click-through searched results and clicked URLs logs were found. There-fore, for inferring user search goals, it is more efficient to analyze the feedback sessions than to analyze the search results and clicked URLs.

III.II CONVERSION OF FEEDBACK SESSIONS TO PSEUDO-DOCUMENTS

Steps to build pseudo-document:

III.II.I. **Representation of the URLs in the Feedback Session**.

→First improve the URLs with additional textual contents by extracting the titles and snippets of the re-turned URLs appearing in the feedback session. In such a way, each URL in the feedback session is represented by a small text paragraph that consists of that URLs title and snippet. After that, some textual processes are implemented to those text paragraphs, such as transforming all of the letters to lowercases, stemming and removing stop words. Finally, each URL's title and snippet are represented by a Term Frequency-Inverse.

Document Frequency (TF-IDF) vector, respectively, as in

Ti = [tw1, tw2, tw3...tw$_n$] ^{T}

Si= [sw1, sw2, sw3…swn] ^{T} _______ (1)

Where Ti and Si are the TF-IDF vectors of the URLs' Title and Snippet respectively.

wj= (1,2,3,………n) is the jth term

appearing in the enriched URLs. Here, a "term" is defined as a word or a number in the dictionary of document collections.

twj and swj represent the TF-IDF value of the jth term in the URL's title and snippet, respectively. Considering that each URLs' titles and snippets have different significances, we represent the each enriched URL by the weighted sum of Tui and Sui namely

Ri = wt.Ti + st.Si = [fw1, fw2… fwn] ^{T}
_____________ (2)

Where R$_{,}$ means the feature representation of the Ith URL in the feedback session, and wt and st are the weights of the titles and the snippets, respectively.

III.II.II **Pseudo-Document formation**

We propose an optimization method to combine clicked and un-clicked URLs in the feedback session to obtain a feature representation.

Let R be the feature representation of a feedback session, and (w) be the value for the term w.

Let C = (m=1, 2, 3… M), and
UC = (l=1, 2, 3… L);

Let R be the feature representations of the clicked and un-clicked URLs in this feedback session, respectively.

Let C and UC be the values for the term w in the vectors. We want to obtain such S that the sum of the distances between S and each C is minimized and the sum of the distances between S and each UC is maximized. Based on the assumption that the terms in the vectors are independent, we can perform optimization on each dimension independently, as shown in below equation

S = [ff(w1) , ff(w2), … ff(wn),…….]
____________ (3)

$$\text{RS (w) = argml} \sum_{M=1}^{n} \left(\mathbf{S(w) - C(w)}\right)\mathbf{2} \ - \gamma \sum_{L=0}^{n} \left(\mathbf{s(w) - Uc(w)}\right)_2 \text{ ____________ (4)}$$

ɣ is a parameter balancing the importance of clicked and un-clicked URLs. When ɣ in (4) is 0, un-clicked URLs are not taken into account. On the other hand, if √ is too big, un-clicked URLs will dominate the value of Uc. In this project, we set √ to be 0.5.

5.3 Clustering Pseudo Document

S in equation (3) and (4), each feedback session is represented by a pseudo document and the feature representation of the pseudo-document is Rs. The similarity between two pseudo documents is computed as the cosine score of Rsi and Rsj, as follows:

$$Sim\ i,j = \frac{cos[Rsi, Rsj]}{|Rsi|. |Rsj|}$$

And distance between two feedback sessions is calculated by

DiSij = 1-Simij

To cluster pseudo documents K-means clustering is used which is very simple and effective. To check the optimal values of clustering we have an evaluation criterion.

IV. BISECTING ALGORITHM

For Bisecting algorithm you must cluster documents using k-means algorithm and then on the result of k-means algorithm.

The idea is iteratively splitting the points into 2 parts.

In other words, build a random binary tree where each splitting (a node with two children) corresponds to splitting the points that begins with a points cloud.

Compute its cancroids (b array center) w

Select randomly a point cL among the points of the cloud

Construct point cR as the symmetric point of cL

When compared to w (the segment cL→w is the same as w→cR)

Separate the points of your cloud in two, the ones closest to cR belong to the sub cloud R, and the ones closest to cL belongs to the sub cloud L

Reiterate for the sub clouds R and L

Discard the random points after using them keep the centroids of all the sub clouds. Stop at point when the sub clouds contain exactly one point

V. EVALUATION CRITERION

V.I AVERAGE PRECISION

A possible evaluation criterion is the average precision (AP) which evaluates according to user implicit feedbacks. AP is the average of precisions which is computed at the point of each relevant document in the ranked sequence, shown in

AP = (6)

Where N is the number of relevant (or clicked) retrieved documents, r is the rank, N is the total number of retrieved documents, rel(r) 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.

V.II VOTED AP (VAP)

It is calculated for purpose of restructuring of search results classes i.e. different clustered results classes. It is same as AP and calculated for class which having more clicks.

V.III RISK

It is the AP of the class including more clicks? There should be a risk to avoid classifying search results into too many classes by error. So we propose the Risk as:

$$\text{Risk} = \frac{\sum_{i=1}^{n} (i < j)\ \text{dij}}{C^2 m}$$

V.IV CLASSIFIED AP (CAP)

VAP is extended to CAP by introducing combination of VAP and Risk.

Classified AP can be calculated by using the formula, as follows:

CAP= VAP X (1- Risk)

6. **CONCLUSION**

The proposed system can be used to improve discovery of user search goals for a similar query by using bisecting algorithm for clustering user feedback sessions represented by pseudo-documents. By using proposed system, the inferred user search goals can be used to restructure web search results. So, users can find exact information quickly and very efficiently. The discovered clusters of query can also be used to assist users in web search.

References:

[1] S. Beitzel, E. Jensen, A. Chowdhury, and O. Frieder, “Varying Ap-proaches to Topical Web Query Classification,” ACM SIGIR Conf. Research and Development (SIGIR ’07), pp. 783-784, 2007.

[2] B. Poblete and B.-Y Ricardo, “Query-Sets: Using Implicit Feedback and Query Patterns to Organize Web Documents,” Proc. 17th Int’l Conf.

[3] 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.

[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] 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.

[6] 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.