www.ijecs.in
International Journal Of Engineering And Computer Science ISSN:2319-7242
Volume 4 Issue 3 March 2015, Page No. 10859-10862

Inferring User Search Goals with feedback Sessions using Fuzzy K-Means Algorithm

Mr.Gajanan Patil, Miss.Sonal Patil

G.H.Raisoni Institute of Engineering & Management, Jalgaon, NMU Jalgaon,
Shirsoli Road, Jalgaon, Maharashtra, India.
Patilgajanan20@gmail.com

Assistant Professor, GHRIEM, Shirsoli Road, Jalgaon, NMU Jalgaon,
Maharashtra, India.
Sonalpatil3@gmail.com

Abstract: ***This document shows the concept for a broad topic and ambiguous query, different types of users may have different search goals when they submit the query to the search engine. The inference and analysis of user search goals can be very useful in improving search engine relevance information and user experience. In this paper, we propose a novel approach to infer user search goals by analyzing search engine query logs. First, we propose a framework to search different user search goals for a query by making cluster to the proposed feedback sessions. Feedback sessions are constructed from user click-through logs i.e. user response and can efficiently reflect the information needs to users. Second, we propose a novel approach to create pseudo-documents to better represent the feedback sessions for clustering. Finally, we propose a new criterion Classified Average Precision (CAP) to calculate the performance of inferring user search goals. Experimental results are presented using user click-through logs from a commercial search engine to check the effectiveness of our proposed methods.***

Keywords: User search goals, feedback sessions, pseudo-documents, restructuring search results, and classified average precision, Fuzzy K-Means Clustering.

1. Introduction

In web search application, queries are submitted to the search engine to represent the information needs to the users. However, sometimes queries may not exactly represent the users' specific information needs since many ambiguous queries may cover a broad topic and different users may want to get information on different aspects when they submit the same query. For example, when the query "the Sun" is submitted to a search engine, some users want to locate the homepage of a United Kingdom newspaper, while some others want to learn the natural knowledge of the Sun, as shown in Fig 1. Therefore, it is necessary to accept the different user search goals in information retrieval. We define user search goals as the information on different aspects of a query that user groups want to obtain. Information need is a user's particular desire to obtain information to satisfy his/her need. User search goals can be considered as the clusters of information needs for a query[1]. The inference and analysis of user search goals can have a lot of advantages in improving search engine relevance and user experience. Some benefits are summarized as follows. First, we can restructure the web search result as per the user search goals by grouping the search result with the same search goal; thus, users with different search goals can easily find what they want. Second, user search goals represented by some keywords can be utilized in the query recommendation; thus, the suggested queries can help users to form their queries more precisely. Third, the distributions of user search goals can also be useful in applications such as reranking web search results that contain different user search goals.

Figure.1. Different user search goals and their distributions for the query "the sun".

This shows the number of user search goals for the query and depicting each goal with the some keywords automatically. First, we proposed a novel approach to infer the user search goals for a query by clustering our proposed feedback system. The feedback session is defined as the series of both clicked and unclicked URLs and end with the last URL that was clicked from user through logs. Then, we proposed the optimization method to map the feedback sessions to pseudo document which can efficiently reflect the user information. At last, we cluster these pseudo documents to infer the user search goals and depict them with some keyword. Since the evaluation of clustering is also an important problem, we also propose a

novel criterion that is Classified Average Precision (CAP) to evaluate the performance of restructured web search result.

2. Feedback Session

Generally, a session for web search is a series of successive queries to satisfy a single information need and some clicked search results. In this paper, we focus on inferring user search goals for a particular query. Therefore, the single session containing only one query is introduced, which distinguishes from the conventional session. While, the feedback session in this paper is based on single session, although it can be extended to the whole session[2].

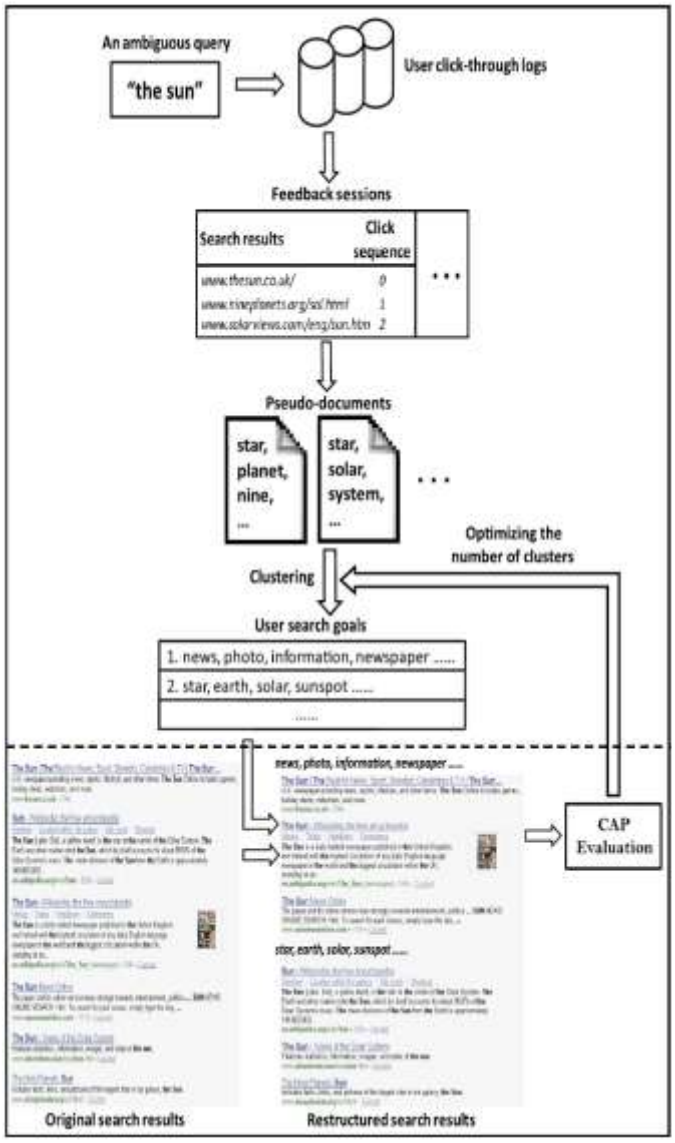

Figure.2. System Architecture.

3. Map Feedback Sessions to Pseudo-Document

Since feedback sessions vary a lot for different click-through and queries, it is unsuitable to directly use feedback sessions for inferring user search goals. There is need of some representation method to describe the feedback sessions in a more efficient and convenient way. There are many kind of features that can represent the feedback session.

Search results	Click sequence
www.thesun.co.uk/	0
www.nineplanets.org/sol.html	1
www.solarviews.com/eng/sun.htm	2
en.wikipedia.org/wiki/Sun	0
www.thesunmagazine.org/	0
www.space.com/sun/	0
en.wikipedia.org/wiki/The_Sun_(newspaper)	3
imagine.gsfc.nasa.gov/docs/science/know_l1/sun.html	0
www.nasa.gov/worldbook/sun_worldbook.html	0
www.enchantedlearning.com/subjects/astronomy/sun/	0

Figure.3A. A feedback session in a single session. "0" in click sequence means "unclicked." All the 10 URLs construct a single session. The URLs in the rectangular box construct a feedback session.

Fig.2 shows that search results are the URLs returned by the search engine when a query "the Sun" is submitted, and 0 represent "Unclicked." In the clicked sequence. The binary vector [0110001] can be used to represent the feedback session, where "1" represents "clicked" and "0" represents "unclicked." However, since different feedback sessions have different numbers of URLs, the binary vectors of different feedback sessions may have different dimensions. Moreover, binary vector representation is not informative enough to tell the contents of user search goals. Therefore, it is improper to use methods such as the binary vectors and new methods are needed to represent feedback sessions. Fig.3B shows a popular binary vector method to represent a feedback session. In this paper, we propose a novel way to map feedback sessions to pseudo-documents, as illustrated in Fig 3.The building of a Pseudo-document includes two steps. They are described in following[3].

3.1 Representing the URLs in the feedback session.

In the first step, we first enrich the URLs with additional textual contents by extracting the titles and snippets of the returned URLs appearing in the feedback session. In this way, each URL in feedback session is represented by a small text paragraph that consists of its title and snippet. Then, some textual processes are implemented to those text paragraphs, such as transforming all letters to lowercase, stemming and removing stop words.

Finally, each URLs title and snippet are represented by Term frequency Inverse Document Frequency (TF-IDF) vector respectively as in

$$\mathbf{T}ui = [t\omega 1,\ t\ \omega 2 \ldots t\ \omega n].\hat{}T,$$
$$\mathbf{S}ui = [s\omega 1,\ s\ \omega 2 \ldots . s\ \omega n].\hat{}T, \qquad (1).$$

Search results	Click sequence	Binary vector
www.thesun.co.uk/	0	0
www.nineplanets.org/sol.html	1	1
www.solarviews.com/eng/sun.htm	2	1
en.wikipedia.org/wiki/Sun	0	0
www.thesunmagazine.org/	0	0
www.space.com/sun/	0	0
en.wikipedia.org/wiki/The_Sun_(newspaper)	3	1

Figure.3B. The binary vector representation of a feedback session

Where T*ui* and S*ui* are the TF-IDF vectors of the URL's title and snippet, respectively. *Ui* means the *i*th URL in the feedback session. And *ωj (j=1, 2… n)* is the *jth* term appearing in the enriched URLs[4]. Here, a "term" is defined as a word or a number in the dictionary of document collections. t*wj* and s*wj* represent the TF-IDF value of the *j*th term in the URL's title and snippet, respectively. Considering that URLs' titles and snippets have different significance, we represent the enriched URL by the weighted sum of T*ui* and S*ui* , namely

$$Fui = \omega t Tui + \omega s Sui = [f\omega 1, f\omega 2 \ldots . f\omega n] \hat{} T. \qquad (2).$$

Where F*ui* means the features representation of the *i*th URLs in the feedback session, and ωt and ωs are the weights of the titles and the snippets, respectively. We set ωs to be 1 at first.

Then, we stipulate that the titles should be more significant than the snippets. Therefore, the weight of the titles should be higher and we set ωt to be 2 in this paper. We also tried to set ωt to be 1.5, the results were similar. Based on (2), the feature representation of the URLs in the feedback session can be obtained. It is worth noting that although T*ui* and S*ui* are TF-IDF features, F*ui* is not a TF-IDF feature. This is because the normalized TF feature is relative to the documents and therefore it cannot be aggregated across documents[5]. In our cases, each term of F*ui* indicates the importance of a term in the *i*th URL.

4. Evalution Criterion

In order to apply the evaluation method to large-scale data, the single sessions in user click-through logs are used to minimize manual work. Because from user click-through logs, we can get implicit relevance feedbacks, namely "clicked" means relevant and "unclicked" means irrelevant[6]. A possible evaluation criterion is the average precision (AP) which evaluates according to user implicit feedbacks. AP is the average of precisions computed at the point of each relevant document in the ranked sequence, as shown in

$$AP = 1/N + \sum rel(r)\, Rr/r,$$

Where Nþ is the number of relevant (or licked) documents in the retrieved ones, r is the rank, N is total number of retrieved document rel(r) is the binary function on the relevance rank. Rr is the relevance retrieved document of the rank r or less. If the numbers of the clicks in two classes are the same, we select the bigger AP as the VAP. Assume that one user has only one search goal, then ideally all the clicked URLs in a single session should belong to one class. And a good restructuring of search results should have higher VAP. However, VAP is still an unsatisfactory criterion.

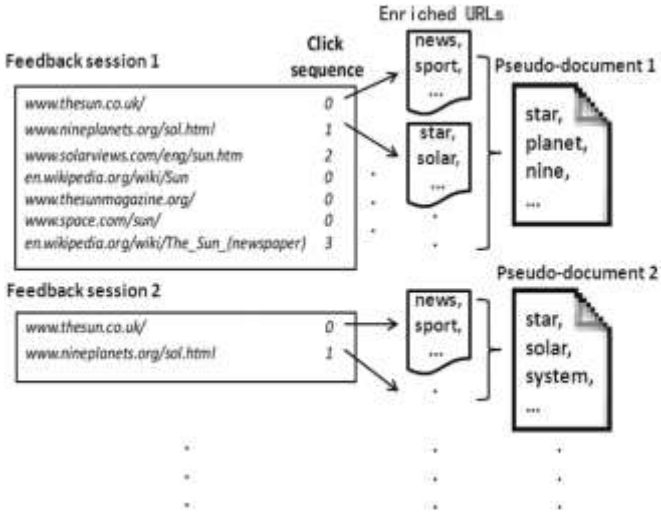

Figure.4 Illustrations for mapping feedback session to pseudo –document.

Considering an extreme case, if each URL in the click session is categorized into one class, VAP will always be the highest value namely 1 no matter whether users have so many search goals or not. Therefore, there should be a risk to avoid classifying search results into too many classes by error.

Table.4. Cap comparison of three methods for 1,720 queries

Method	Mean Average VAP	Mean Average Risk	Mean Average CAP
Our Method	0.755	0.224	0.632
Method I	0.680	0.196	0.584
Method II	0.742	0.243	0.611

5. Fuzzy K-Means Algorithm

The fuzzy k-means clustering (FKM) algorithm performs the partition step iteratively and new cluster representative generation step until convergence. An iterative process with extensive computations is usually required to generate a set of cluster representatives. The convergence of FKM is usually much less than that of standard K-means clustering. Some methods are available to speed up hard k-means clustering developed a filtering algorithm on a kd-tree to speed up the generation of new cluster center. In fuzzy clustering, every point has a degree of belonging to clusters, as in fuzzy logic, rather than belonging completely to just one cluster[8]. Thus, points on the edge of a cluster may be *in the cluster* to a lesser degree than points in the center of cluster. An overview and comparison of different fuzzy clustering algorithms is available. Any point *x* has a set of coefficients giving the degree of being in the *k*th cluster $w_k(x)$. With fuzzy K-means, the centroid of a cluster is the mean of all points, weighted by their degree of belonging to the cluster:

$$c_k = \frac{\sum_x w_k(x)^m x}{\sum_x w_k(x)^m}.$$

The degree of belonging, $w_k(x)$, is related inversely to the distance from x to the cluster center as calculated on the previous pass. It also depends on a parameter m that controls how much weight is given to the closest center. The fuzzy *K*-means algorithm is very similar to the *k*-means algorithm[9]. The clusters produced by the k-means procedure are sometimes called "hard" or "crisp" clusters, since any feature vector x either is or is not a member of a particular cluster. This is in contrast to "soft" or "fuzzy" clusters, in which a feature vector x can have a degree of membership in each cluster. The fuzzy-k-means procedure of Dunn and Bezdek allows each feature vector x to have a degree of membership in Cluster i:

- Make initial guesses for the means $\mathbf{m}_1$, $\mathbf{m}_2$,..., $\mathbf{m}_k$.
- Until there are no changes in any mean:
- Use the estimated means to find the degree of membership u(j,i) of $\mathbf{x}_j$ in Cluster i; for example, if a(j,i) = exp(- $\| \mathbf{x}_j - \mathbf{m}_i \|^2$), one might use u(j,i) = a(j,i) / sum_j a(j,i)
- For i from 1 to k.Replace $\mathbf{m}_i$ with the fuzzy mean of all of the examples for Cluster i –

$$\mathbf{m_i} = \frac{\sum_j u(j,i)^2 \mathbf{x_j}}{\sum_j u(j,i)^2}$$

- end_for
- end_until

6. Conclusion

For each query, the running time depends on the number of feedback sessions. However, the dimension is not very high. Therefore, the running time is usually short. In reality, our approach can discover user search goals for some popular queries offline at first. Then, when users submit one of the queries, the search engine can return the results that are categorized into different groups according to user search goals online. Thus, users can find what they want conveniently. The algorithm is more effective than other and required less time to reflect the user information.

References

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

[2] 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.

[3] 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.

[4] S. Beitzel, E. Jensen, A. Chowdhury, and O. Frieder, "Varying Approaches to Topical Web Query Classification," Proc. 30th Ann Int'l ACM SIGIR Conf. Research and Development (SIGIR '07), pp. 783-784, 2007.

[5] H. Cao, D. Jiang, J. Pei, Q. He, Z. Liao, E. Chen, and H. Li,"Context-Aware Query Suggestion by Mining Click-Through,"Proc. 14th ACM SIGKDD Int'l Conf.Knowledge Discovery and Data Mining (SIGKDD '08), pp.875-883, 2008.

[6] 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.

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

[8] 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.

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

Author Profile

Gajanan Patil received the B.E(I.T) and studing in M.E in Computer Science & engineering from G.H.Raisoni Institute of engineering & Management in 2013 and 2015,respectively.