A New Algorithm for Inferring User Search Goals with Feedback Sessions

Zheng Lu, *Student Member*, *IEEE*, Hongyuan Zha, Xiaokang Yang, *Senior Member*, *IEEE*, Weiyao Lin, *Member*, *IEEE*, and Zhaohui Zheng

Abstract—For a broad-topic and ambiguous query, different users may have different search goals when they submit it to a search engine. The inference and analysis of user search goals can be very useful in improving search engine relevance 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 discover different user search goals for a query by clustering the proposed feedback sessions. Feedback sessions are constructed from user click-through logs and can efficiently reflect the information needs of users. Second, we propose a novel approach to generate pseudo-documents to better represent the feedback sessions for clustering. Finally, we propose a new criterion "Classified Average Precision (CAP)" to evaluate the performance of inferring user search goals. Experimental results are presented using user click-through logs from a commercial search engine to validate the effectiveness of our proposed methods.

Index Terms—User search goals, feedback sessions, pseudo-documents, restructuring search results, classified average precision

1 Introduction

In web search applications, queries are submitted to search engines to represent the information needs of users. However, sometimes queries may not exactly represent 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 and potential to capture 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. The inference and analysis of user search goals can have a lot of advantages in improving search engine relevance and user experience. Some advantages are summarized as follows. First, we can restructure web search results [6], [18], [20] according to user search goals by grouping the search results 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 query recommendation [2], [5], [7]; 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.

Due to its usefulness, many works about user search goals analysis have been investigated. They can be summarized into three classes: query classification, search result reorganization, and session boundary detection. In the first class, people attempt to infer user goals and intents by predefining some specific classes and performing query classification accordingly. Lee et al. [13] consider user goals as "Navigational" and "Informational" and categorize queries into these two classes. Li et al. [14] define query intents as "Product intent" and "Job intent" and they try to classify queries according to the defined intents. Other works focus on tagging queries with some predefined concepts to improve feature representation of queries [17]. However, since what users care about varies a lot for different queries, finding suitable predefined search goal classes is very difficult and impractical. In the second class, people try to reorganize search results. Wang and Zhai [18] learn interesting aspects of queries by analyzing the clicked URLs directly from user click-through logs to organize search results. However, this method has limitations since the number of different clicked URLs of a query may be small. Other works analyze the search results returned by the search engine when a query is submitted [6], [20]. Since user feedback is not considered, many noisy search results that are not clicked by any users may be analyzed as well. Therefore, this kind of methods cannot infer user search

- Z. Lu and W. Lin are with the Department of Electronic Engineering, Shanghai Jiao Tong University, SEIEE buildings, 800 Dong Chuan Road, Shanghai 200240, China. E-mail: {luzheng, wylin}@sjtu.edu.cn.
- H. Zha is with the School of Computational Science and Engineering, College of Computing, Georgia Institute of Technology, Atlanta, GA 30332. E-mail: zha@cc.gatech.edu.
- X. Yang is with the Shanghai Key Laboratory of Digital Media Processing and Communication, Department of Electronic Engineering, Institute of Image Communication and Network Engineering, Shanghai Jiao Tong University, 800 Dong Chuan Road, Shanghai 200240, China. E-mail: xkyang@sjtu.edu.cn.
- Z. Zheng is with Yahoo! Labs Beijing, Floor 10, Qinghua Tongfang building, 1 Wang Zhuang Road, Beijing 100083, China. E-mail: zhaohui@yahoo-inc.com.

Manuscript received 18 Aug. 2010; revised 3 Nov. 2011; accepted 23 Nov. 2011; published online 5 Dec. 2011.
For information on obtaining reprints of this article, please send e-mail to: tkde@computer.org, and reference IEEECS Log Number TKDE-2010-08-0455.
Digital Object Identifier no. 10.1109/TKDE.2011.248.

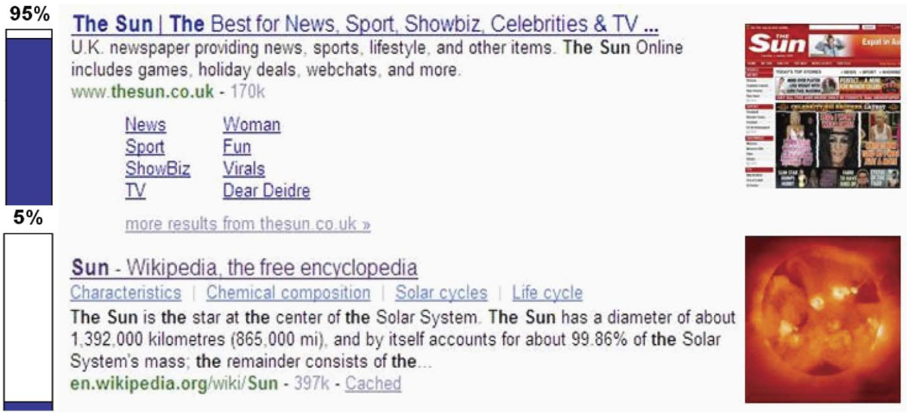


Fig. 1. The examples of the different user search goals and their distributions for the query "the sun" by our experiment.

goals precisely. In the third class, people aim at detecting session boundaries. Jones and Klinkner [11] predict goal and mission boundaries to hierarchically segment query logs. However, their method only identifies whether a pair of queries belong to the same goal or mission and does not care what the goal is in detail.

In this paper, we aim at discovering the number of diverse user search goals for a query and depicting each goal with some keywords automatically. We first propose a novel approach to infer user search goals for a query by clustering our proposed feedback sessions. The feedback session is defined as the series of both clicked and unclicked URLs and ends with the last URL that was clicked in a session from user click-through logs. Then, we propose a novel optimization method to map feedback sessions to pseudo-documents which can efficiently reflect user information needs. At last, we cluster these pseudo-documents to infer user search goals and depict them with some keywords. Since the evaluation of clustering is also an important problem, we also propose a novel evaluation criterion classified average precision (CAP) to evaluate the performance of the restructured web search results. We also demonstrate that the proposed evaluation criterion can help us to optimize the parameter in the clustering method when inferring user search goals.

To sum up, our work has three major contributions as follows:

- We propose a framework to infer different user search goals for a query by clustering feedback sessions. We demonstrate that clustering feedback sessions is more efficient than clustering search results or clicked URLs directly. Moreover, the distributions of different user search goals can be obtained conveniently after feedback sessions are clustered.
- We propose a novel optimization method to combine the enriched URLs in a feedback session to form a pseudo-document, which can effectively reflect the information need of a user. Thus, we can tell what the user search goals are in detail.
- We propose a new criterion CAP to evaluate the performance of user search goal inference based on restructuring web search results. Thus, we can determine the number of user search goals for a query.

The rest of the paper is organized as follows: The framework of our approach is presented in Section 2. The proposed feedback sessions and their representation namely pseudo-documents are described in Section 3. Section 4 describes the proposed method to infer user search goals. The evaluation criterion CAP is proposed in Section 5. Section 6 shows the experimental results and analysis. Section 7 reviews several related works and Section 8 concludes the paper.

2 Framework of Our Approach

Fig. 2 shows the framework of our approach. Our framework consists of two parts divided by the dashed line.

In the upper part, all the feedback sessions of a query are first extracted from user click-through logs and mapped to pseudo-documents. Then, user search goals are inferred by clustering these pseudo-documents and depicted with some keywords. Since we do not know the exact number of user search goals in advance, several different values are tried and the optimal value will be determined by the feedback from the bottom part.

In the bottom part, the original search results are restructured based on the user search goals inferred from the upper part. Then, we evaluate the performance of restructuring search results by our proposed evaluation criterion CAP. And the evaluation result will be used as the feedback to select the optimal number of user search goals in the upper part.

3 Representation of Feedback Sessions

In this section, we first describe the proposed feedback sessions and then we introduce the proposed pseudo-documents to represent feedback sessions.

3.1 Feedback Sessions

Generally, a session for web search is a series of successive queries to satisfy a single information need and some clicked search results [11]. 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. Meanwhile, the feedback session in this paper is based on a single session, although it can be extended to the whole session.

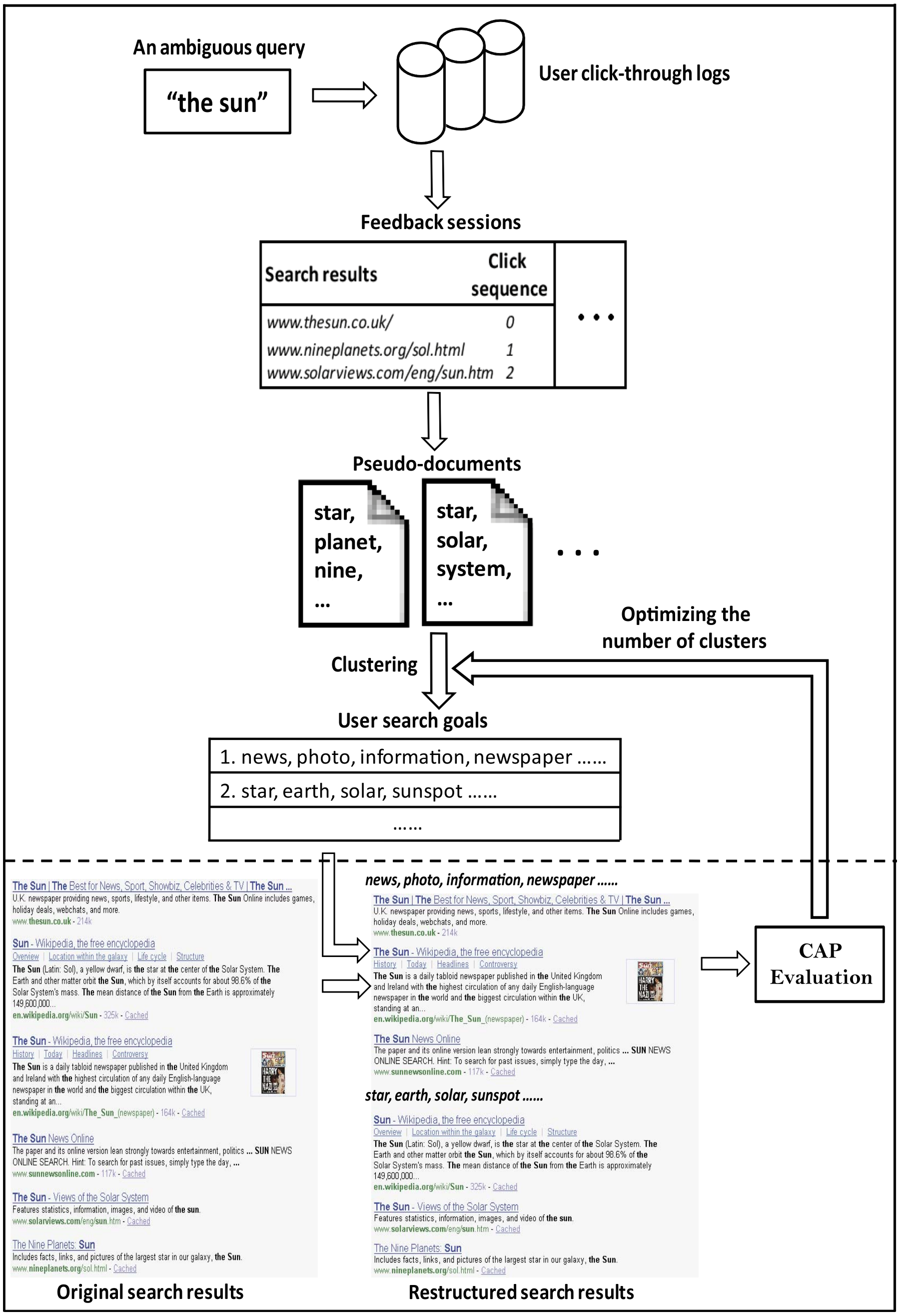


Fig. 2. The framework of our approach.

The proposed feedback session consists of both clicked and unclicked URLs and ends with the last URL that was clicked in a single session. It is motivated that before the last click, all the URLs have been scanned and evaluated by users. Therefore, besides the clicked URLs, the unclicked ones before the last click should be a part of the user feedbacks. Fig. 3 shows an example of a feedback session and a single session. In Fig. 3, 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 "unclicked." The single session includes all the 10 URLs in Fig. 3, while the feedback session only includes the seven URLs in the rectangular box. The seven URLs consist of three clicked URLs and four unclicked URLs in this example. Generally speaking, since users will scan the URLs one by one from top to down, we can consider that besides the three clicked URLs, the four unclicked ones in the rectangular box have also been browsed and evaluated by the user and they should reasonably be a part of the user feedback. Inside the feedback session, the clicked URLs tell what users require and the unclicked URLs reflect what users do not care about. It should be noted that the unclicked 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. Moreover, there are plenty of diverse feedback sessions in user click-through logs. Therefore, for inferring user search goals, it is more efficient to analyze the feedback sessions than to analyze the search results or clicked URLs directly.

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

Fig. 3. 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.

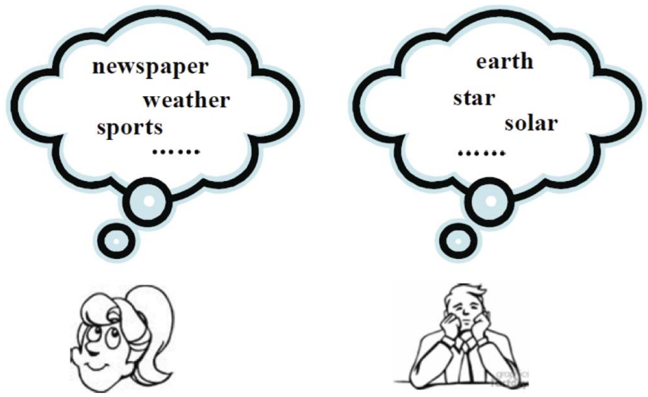


Fig. 5. Goal texts. For a query, different users will have different keywords in their minds. These keywords are vague and have no order. We name them "goal texts," which reflect user information needs.

3.2 Map Feedback Sessions to Pseudo-Documents

Since feedback sessions vary a lot for different click-throughs and queries, it is unsuitable to directly use feedback sessions for inferring user search goals. Some representation method is needed to describe feedback sessions in a more efficient and coherent way. There can be many kinds of feature representations of feedback sessions. For example, Fig. 4 shows a popular binary vector method to represent a feedback session. Same as Fig. 3, search results are the URLs returned by the search engine when the query "the sun" is submitted, and "0" represents "unclicked" in the click 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.

For a query, users will usually have some vague keywords representing their interests in their minds. They use these keywords to determine whether a document can satisfy their needs. We name these keywords "goal texts" as shown in Fig. 5. However, although goal texts can reflect user information needs, they are latent and not expressed explicitly. Therefore, we introduce pseudo-documents as surrogates to approximate goal texts. Thus, pseudo-documents can be used to infer user search goals.

In this paper, we propose a novel way to map feedback sessions to pseudo-documents, as illustrated in Fig. 6. The building of a pseudo-document includes two steps. They are described in the following:

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

Fig. 4. The binary vector representation of a feedback session.

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 a 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 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 [1], respectively, as in

$$\begin{aligned} \mathbf{T}_{u_i} &= [t_{w_1}, t_{w_2}, \ldots, t_{w_n}]^T, \\ \mathbf{S}_{u_i} &= [s_{w_1}, s_{w_2}, \ldots, s_{w_n}]^T, \end{aligned} \tag{1}$$

where $\mathbf{T}_{u_i}$ and $\mathbf{S}_{u_i}$ are the TF-IDF vectors of the URL's title and snippet, respectively. u_i means the ith URL in the feedback session. And $w_j (j = 1, 2, \ldots, n)$ 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. t_{w_j} and s_{w_j} represent the TF-IDF value of the jth term in the URL's title and snippet, respectively. Considering that URLs' titles and snippets have different significances, we represent the enriched URL by the weighted sum of $\mathbf{T}_{u_i}$ and $\mathbf{S}_{u_i}$, namely

$$\mathbf{F}_{u_i} = \omega_t \mathbf{T}_{u_i} + \omega_s \mathbf{S}_{u_i} = [f_{w_1}, f_{w_2}, \ldots, f_{w_n}]^T, \tag{2}$$

where $\mathbf{F}_{u_i}$ means the feature representation of the ith URL in the feedback session, and ω_t and ω_s are the weights of the titles and the snippets, respectively. We set ω_s 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 ω_t to be 2 in this paper. We also tried to set ω_t 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 $\mathbf{T}_{u_i}$ and $\mathbf{S}_{u_i}$ are TF-IDF features, $\mathbf{F}_{u_i}$ 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. In our case, each term of $\mathbf{F}_{u_i}$ (i.e., f_{w_j}) indicates the importance of a term in the ith URL.

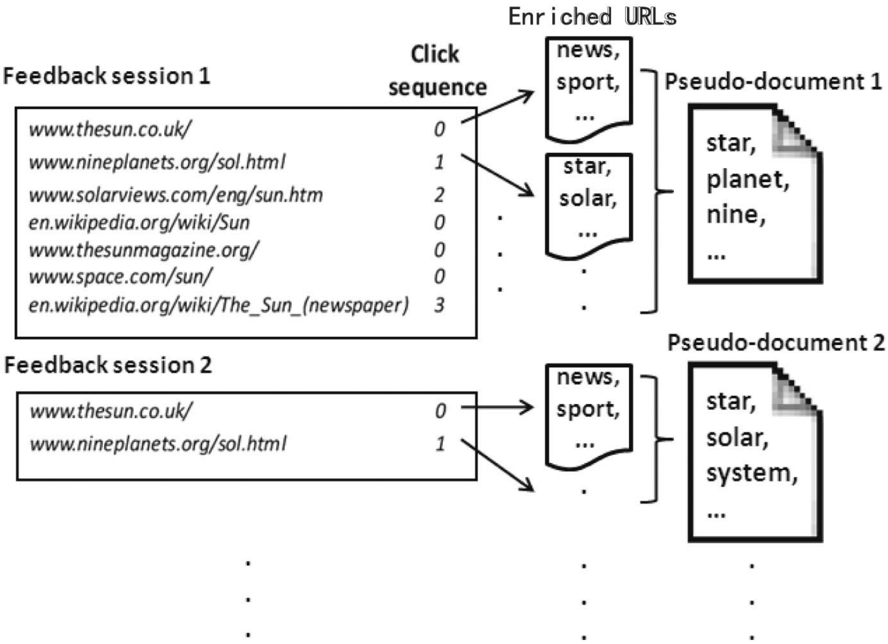


Fig. 6. Illustration for mapping feedback sessions to pseudo-documents.

2) *Forming pseudo-document based on URL representations.* In order to obtain the feature representation of a feedback session, we propose an optimization method to combine both clicked and unclicked URLs in the feedback session. Let $\mathbf{F}_{fs}$ be the feature representation of a feedback session, and $f_{fs}(w)$ be the value for the term w. Let $\mathbf{F}_{uc_m}(m = 1, 2, \ldots, M)$ and $\mathbf{F}_{u\bar{c}_l}(l = 1, 2, \ldots, L)$ be the feature representations of the clicked and unclicked URLs in this feedback session, respectively. Let $f_{uc_m}(w)$ and $f_{u\bar{c}_l}(w)$ be the values for the term w in the vectors. We want to obtain such a $\mathbf{F}_{fs}$ that the sum of the distances between $\mathbf{F}_{fs}$ and each $\mathbf{F}_{uc_m}$ is minimized and the sum of the distances between $\mathbf{F}_{fs}$ and each $\mathbf{F}_{u\bar{c}_l}$ 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

$$\begin{aligned} \mathbf{F}_{fs} &= \left[f_{fs}(w_1), f_{fs}(w_2), \ldots f_{fs}(w_n)\right]^T, \\ f_{fs}(w) &= \arg\min_{f_{fs}(w)} \left\{ \sum_M \left[f_{fs}(w) - f_{uc_m}(w)\right]^2 \right. \\ &\quad \left. - \lambda \sum_L \left[f_{fs}(w) - f_{u\bar{c}_l}(w)\right]^2 \right\}, f_{fs}(w) \in I_c. \end{aligned} \tag{3}$$

Let I_c be the interval $[\mu_{f_{uc}}(w) - \sigma_{f_{uc}}(w), \mu_{f_{uc}}(w) + \sigma_{f_{uc}}(w)]$ and $I_{\bar{c}}$ be the interval $[\mu_{f_{u\bar{c}}}(w) - \sigma_{f_{u\bar{c}}}(w), \mu_{f_{u\bar{c}}}(w) + \sigma_{f_{u\bar{c}}}(w)]$, where $\mu_{f_{uc}}(w)$ and $\sigma_{f_{uc}}(w)$ represent the mean and mean square error of $f_{uc}(w)$ respectively, and $\mu_{f_{u\bar{c}}}(w)$ and $\sigma_{f_{u\bar{c}}}(w)$ represent the mean and mean square error of $f_{u\bar{c}}(w)$, respectively. If $I_c \subseteq I_{\bar{c}}$ or $I_{\bar{c}} \subseteq I_c$, we consider that the user does not care about the term w. In this situation, we set $f_{fs}(w)$ to be 0, as shown in

$$f_{fs}(w) = 0, I_c \subseteq I_{\bar{c}} \text{ or } I_{\bar{c}} \subseteq I_c. \tag{4}$$

λ is a parameter balancing the importance of clicked and unclicked URLs. When λ in (3) is 0, unclicked URLs are not taken into account. On the other hand, if λ is too big, unclicked URLs will dominate the value of $f_{fs}(w)$. In this paper, we set λ to be 0.5.

It is worth noting that people will also skip some URLs because they are too similar to the previous ones. In this situation, the "unclicked" URLs could wrongly reduce the weight of some terms in the pseudo-documents to some extent. However, our method can address this problem. Let us analyze the problem from three cases. Case 1 (the ideal case): one term appears in all the clicked URLs and does not appear in any unclicked ones. In this case, people skip because the unclicked URLs do not contain this important term. The weight of the term in the pseudo-document will be set to the highest value in I_c in (3). Case 2 (the general case): one term appears in both the clicked URLs and a subset of the unclicked ones. In this case, some unclicked URLs are skipped because they are irrelevant and some are skipped because of duplication. The weight of the term will be reduced to some extent; however, it will not be set to zero and it is still included in I_c according to (3). Therefore, skipping because of duplication does not affect too much in this case. Case 3 (the bad case): one term appears in both the clicked URLs and almost all the unclicked ones. In this case, people skip because of duplication. $I_{\bar{c}}$ could contain I_c and the weight of the term will be set to zero according to (4). However, when this case happens, both the clicked and the unclicked URLs are almost about one single subject and the term is no longer distinguishable. Therefore, even if people skip some unclicked URLs because of duplication, our method can still assign reasonable weight of the term in most cases.

Up to now, the feedback session is represented by $\mathbf{F}_{fs}$. Each dimension of $\mathbf{F}_{fs}$ indicates the importance of a term in this feedback session. $\mathbf{F}_{fs}$ is the pseudo-document that we want to introduce. It reflects what users desire and what they do not care about. It can be used to approximate the goal texts in user mind.

4 Inferring User Search Goals by Clustering Pseudo-Documents

With the proposed pseudo-documents, we can infer user search goals. In this section, we will describe how to infer user search goals and depict them with some meaningful keywords.

As in (3) and (4), each feedback session is represented by a pseudo-document and the feature representation of the pseudo-document is $\mathbf{F}_{fs}$. The similarity between two pseudo-documents is computed as the cosine score of $\mathbf{F}_{fs_i}$ and $\mathbf{F}_{fs_j}$, as follows:

$$Sim_{i,j} = \cos\left(\mathbf{F}_{fs_i}, \mathbf{F}_{fs_j}\right) = \frac{\mathbf{F}_{fs_i} \cdot \mathbf{F}_{fs_j}}{|\mathbf{F}_{fs_i}||\mathbf{F}_{fs_j}|}. \tag{5}$$

And the distance between two feedback sessions is

$$Dis_{i,j} = 1 - Sim_{i,j}. \tag{6}$$

We cluster pseudo-documents by K-means clustering which is simple and effective. Since we do not know the exact number of user search goals for each query, we set K to be five different values (i.e., $1, 2, \ldots, 5$) and perform clustering based on these five values, respectively. The optimal value will be determined through the evaluation criterion presented in Section 5.

After clustering all the pseudo-documents, each cluster can be considered as one user search goal. The center point of a cluster is computed as the average of the vectors of all the pseudo-documents in the cluster, as shown in

$$\mathbf{F}_{center_i} = \frac{\sum_{k=1}^{C_i} \mathbf{F}_{fs_k}}{C_i}, \left(\mathbf{F}_{fs_k} \subset Cluster\ i\right), \tag{7}$$

where $\mathbf{F}_{center_i}$ is the ith cluster's center and C_i is the number of the pseudo-documents in the ith cluster. $\mathbf{F}_{center_i}$ is utilized to conclude the search goal of the ith cluster.

Finally, the terms with the highest values in the center points are used as the keywords to depict user search goals. Note that an additional advantage of using this keyword-based description is that the extracted keywords can also be utilized to form a more meaningful query in query recommendation and thus can represent user information needs more effectively.

Moreover, since we can get the number of the feedback sessions in each cluster, the useful distributions of user search goals can be obtained simultaneously. The ratio of the number of the feedback sessions in one cluster and the total number of all the feedback sessions is the distribution of the corresponding user search goal.

5 Evaluation Based on Restructuring Web Search Results

The evaluation of user search goal inference is a big problem, since user search goals are not predefined and there is no ground truth. Previous work has not proposed a suitable approach on this task. Furthermore, since the optimal number of clusters is still not determined when inferring user search goals, a feedback information is needed to finally determine the best cluster number, as shown in Fig. 2. Therefore, it is necessary to develop a metric to evaluate the performance of user search goal inference objectively. Considering that if user search goals are inferred properly, the search results can also be restructured properly, since restructuring web search results is one application of inferring user search goals. Therefore, we propose an evaluation method based on restructuring web search results to evaluate whether user search goals are inferred properly or not. In this section, we propose this novel criterion "Classified Average Precision" to evaluate the restructure results. Based on the proposed criterion, we also describe the method to select the best cluster number.

5.1 Restructuring Web Search Results

Since search engines always return millions of search results, it is necessary to organize them to make it easier for users to find out what they want. Restructuring web search results is an application of inferring user search goals. We will introduce how to restructure web search results by inferred user search goals at first. Then, the evaluation based on restructuring web search results will be described.

The inferred user search goals are represented by the vectors in (7) and the feature representation of each URL in the search results can be computed by (1) and (2). Then, we can categorize each URL into a cluster centered by the inferred search goals. In this paper, we perform categorization by choosing the smallest distance between the URL vector and user-search-goal vectors. By this way, the search results can be restructured according to the inferred user search goals.

5.2 Evaluation 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. A possible evaluation criterion is the average precision (AP) [1] 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

$$AP = \frac{1}{N^+} \sum_{r=1}^{N} rel(r) \frac{R_r}{r}, \tag{8}$$

where N^+ is the number of relevant (or clicked) documents in the retrieved ones, 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 R_r is the number of relevant retrieved documents of rank r or less. For example, Fig. 7a is a single session with user's implicit feedback and we can compute AP as: $\frac{1}{4} \times (\frac{1}{2} + \frac{2}{3} + \frac{3}{7} + \frac{4}{9}) = 0.510$. However, AP is not suitable for evaluating the restructured or clustered searching results. The proposed new criterion for evaluating restructured results is described in the following.

As shown in Fig. 7b, the URLs in the single session are restructured into two classes where the un-boldfaced ones in Fig. 7a are clustered into class 1 and boldfaced ones are clustered into class 2. We first introduce "Voted AP (VAP)" which is the AP of the class including more clicks namely

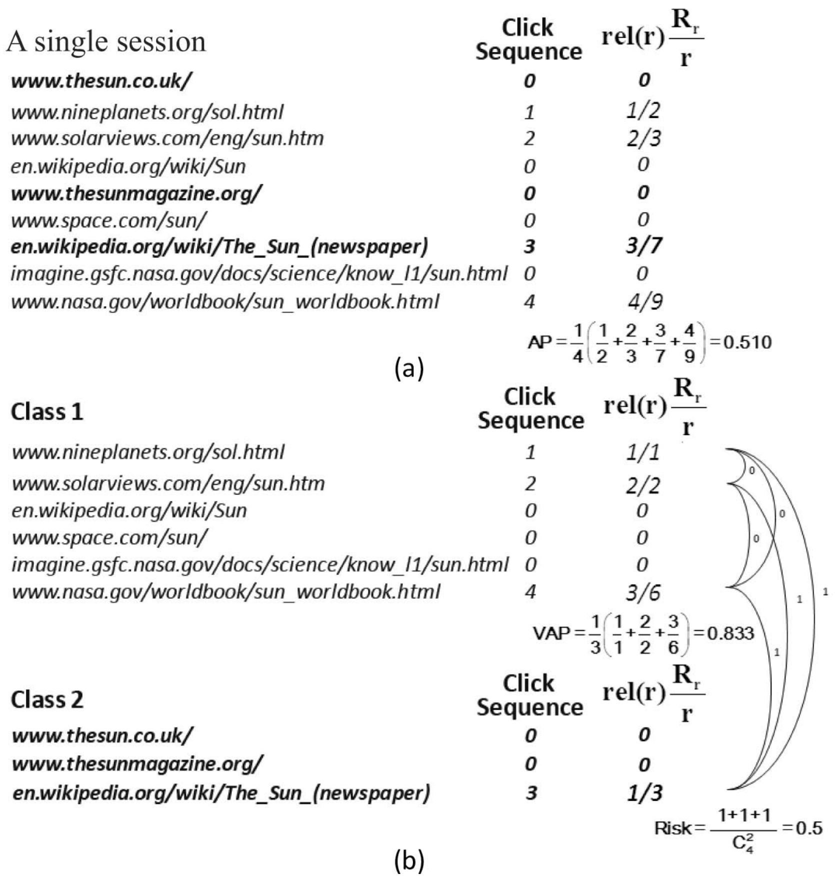


Fig. 7. Illustration for the calculation of AP, VAP, and Risk.

votes. For example, the VAP of the restructured search results in Fig. 7b is the AP of class 1, calculated by: $VAP = \frac{1}{3} \times (\frac{1}{1} + \frac{2}{2} + \frac{3}{6}) = 0.833$. 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. 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. We propose the risk as follows:

$$Risk = \frac{\sum_{i,j=1(i<j)}^{m} d_{ij}}{C_m^2}. \tag{9}$$

It calculates the normalized number of clicked URL pairs that are not in the same class, where m is the number of the clicked URLs. If the pair of the ith clicked URL and the jth clicked URL are not categorized into one class, d_{ij} will be 1; otherwise, it will be 0. $C_m^2 = \frac{m(m-1)}{2}$ is the total number of the clicked URL pairs. In the example of Fig. 7b, the lines connect the clicked URL pairs and the values of the line reflect whether the two URLs are in the same class or not. Then, the risk in Fig. 7b can be calculated by: $Risk = \frac{3}{6} = \frac{1}{2}$. Based on the above discussions, we can further extend VAP by introducing the above Risk and propose a new criterion "Classified AP," as shown below

$$CAP = VAP \times (1 - Risk)^{\gamma}. \tag{10}$$

From (10), we can see that CAP selects the AP of the class that user is interested in (i.e., with the most clicks/votes) and takes the risk of wrong classification into account. And γ is used to adjust the influence of $Risk$ on CAP, which can be learned from training data. Finally, we utilize CAP to evaluate the performance of restructuring search results.

Considering another extreme case, if all the URLs in the search results are categorized into one class, $Risk$ will always be the lowest namely 0; however, VAP could be very low. Generally, categorizing search results into less clusters will induce smaller $Risk$ and bigger VAP, and more clusters will result in bigger $Risk$ and smaller VAP. The proposed CAP depends on both of $Risk$ and VAP.

6 EXPERIMENTS

In this section, we will show experiments of our proposed algorithm. The data set that we used is based on the click-through logs from a commercial search engine collected over a period of two months, including totally 2,300 different queries, 2.5 million single sessions and 2.93 million clicks. On average, each query has 1,087 single sessions and 1,274 clicks. However, these queries are chosen randomly and they have totally different click numbers. Excluding those queries with less than five different clicked URLs, we still have 1,720 queries. Before using the data sets, some preprocesses are implemented to the click-through logs including enriching URLs and term processing.

In our approach, we have two parameters to be fixed: K in K-means clustering and γ in (10). When clustering feedback sessions of a query, we try five different $K(1, 2, \ldots, 5)$ in K-means clustering. Then, we restructure the search results according to the inferred user search goals and evaluate the performance by CAP, respectively. At last, we select K with the highest CAP.

Before computing CAP, we need to determine γ in (10). We select 20 queries and empirically decide the number of user search goals of these queries. Then, we cluster the feedback sessions and restructure the search results with inferred user search goals. We tune the parameter γ to make CAP the highest when K in K-means accord with what we expected for most queries. Based on the above process, the optimal γ is from 0.6 to 0.8 for the 20 queries. The mean and the variance of the optimal γ are 0.697 and 0.005, respectively. Thus, we set γ to be 0.7. Moreover, we use another 20 queries to compute CAP with the optimal γ (0.7) and the result shows that it is proper to set γ to be 0.7.

In the following, we will first give intuitive results of discovering user goals to show that our approach can depict user search goals properly with some meaningful words. Then, we will give the comparison between our method and the other two methods in restructuring web search results.

6.1 Intuitive Results of Inferring User Search Goals

We infer user search goals for a query by clustering its feedback sessions. User search goals are represented by the center points of different clusters. Since each dimension of the feature vector of a center point indicates the importance of the corresponding term, we choose those keywords with the highest values in the feature vector to depict the content of one user search goal.

Table 1 gives some examples of depicting user search goals with four keywords that have the highest values in those feature vectors. From these examples, we can get intuitive results of our search goal inference. Taking the query "lamborghini" as an example, since CAP of the restructured search results is the highest when ($K = 3$), there are totally three clusters (i.e., three lines) corresponding to "lamborghini" and each cluster is represented by four keywords. From the keywords "car, history, company, overview," we can find that this part of users are interested in the history of Lamborghini. From the keywords "new, auto, picture, vehicle," we can see that other users want to retrieve the pictures of new Lamborghini cars. From the keywords "club, oica, worldwide, Lamborghiniclub," we can find that the rest of the users are interested in a Lamborghini club. We can find that the inferred user search goals of the other queries are also meaningful. This confirms that our approach can infer user search goals properly and depict them with some keywords meaningfully.

6.2 Object Evaluation and Comparison

In this section, we will give the objective evaluation of our search goal inference method and the comparison with other two methods.

TABLE 1
Abstracted Keywords Used to Depict User Search Goals for Some Ambiguous Queries

Query	Four keywords to depict user search goals
earth	google, map, wikipedia, planet
	planet, solar, system, nineplanet
	nasa, science, gov, nineplanet
graffiti	art, wall, writing, free
	game, yahoo, art, play
india	map, city, region, information
	travel, information, welcome, land
lamborghini	car, history, company, overview
	new, auto, picture, vehicle
	club, oica, worldwide, lamborghiniclub
sex on the beach	photo, vh1, gallery, cocktail
	recipe, vodka, cocktail, drink
	demeter, fragrance, cocktail, perfume
the sun	news, photo, information, newspaper
	star, earth, solar, sunspot

Three methods are compared. They are described as follows:

- Our proposed method clusters feedback sessions to infer user search goals.
- Method I clusters the top 100 search results to infer user search goals [6], [20]. First, we program to automatically submit the queries to the search engine again and crawl the top 100 search results including their titles and snippets for each query. Then, each search result is mapped to a feature vector according to (1) and (2). Finally, we cluster these 100 search results of a query to infer user search goals by K-means clustering and select the optimal K based on CAP criterion.
- Method II clusters different clicked URLs directly [18]. In user click-through logs, a query has a lot of different single sessions; however, the different clicked URLs may be few. First, we select these different clicked URLs for a query from user click-through logs and enrich them with there titles and snippets as we do in our method. Then, each clicked URL is mapped to a feature vector according to (1) and (2). Finally, we cluster these different clicked URLs directly to infer user search goals as we do in our method and Method I.

In order to demonstrate that when inferring user search goals, clustering our proposed feedback sessions are more efficient than clustering search results and clicked URLs directly, we use the same framework and clustering method. The only difference is that the samples these three methods cluster are different. Note that in order to make the format of the data set suitable for Method I and Method II,

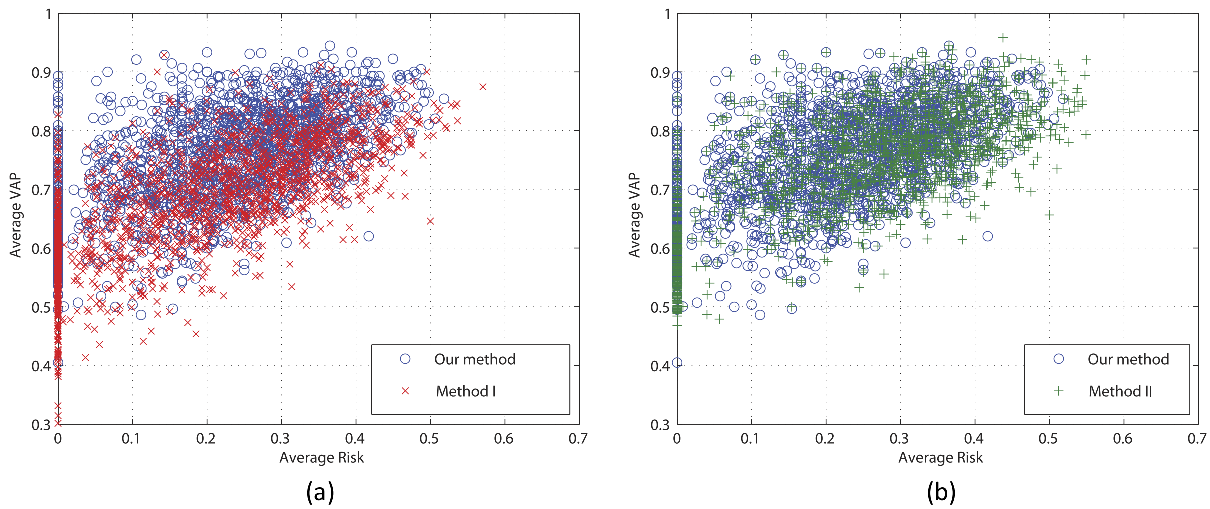


Fig. 8. Comparison of three methods for 1,720 queries. Each point represents the average $Risk$ and VAP of a query when evaluating the performance of restructuring the search results.

some data reorganization is performed to the data set. The performance evaluation and comparison are based on the restructuring web search results.

As shown in Fig. 8, we compare three methods for all the 1,720 queries. Fig. 8a compares our method with Method I and Fig. 8b compares ours with Method II. $Risk$ and VAP are used to evaluate the performance of restructuring search results together. Each point in Fig. 8 represents the average $Risk$ and VAP of a query. If the search results of a query are restructured properly, $Risk$ should be small and VAP should be high and the point should tend to be at the top left corner. We can see that the points of our method are closer to the top left corner comparatively.

We compute the mean average VAP, $Risk$, and CAP of all the 1,720 queries as shown in Table 2. We can see that the mean average CAP of our method is the highest, 8.22 and 3.44 percent higher than Methods I and II respectively. The results of Method I are lower than ours due to the lack of user feedbacks. However, the results of Method II are close to ours. This is because a lot of queries are actually not ambiguous and thus the average improvement of our method for all the 1,720 queries is not very significant. Statistically, our method is better than Method I for 81.8 percent queries in totally 1,720 queries and better than Method II for 69.5 percent queries.

In order to further compare our method with the other two methods, we test the 100 most ambiguous queries. The results are shown in Fig. 9. We can see from Fig. 9 that the points of our method are much closer to the top left corner. The average CAPs of each query of the three methods are shown in Fig. 10. It is obvious that our method usually has the highest average CAP.

Table 3 represents the mean average VAP, $Risk$, and CAP of these 100 queries. It turns out that our proposed method has the highest mean average CAP, which is significantly higher than the other two methods by 36.2 and 14.6 percent. Statistically, our method is better than Method I for 100 percent queries in these 100 queries and better than Method II for 88 percent queries.

6.3 Analyze the Advantages of Clustering Feedback Sessions

In this section, we will give some intuitive explanation showing why clustering feedback sessions namely pseudo-documents is better than the other two methods when inferring user search goals. With the introduction of feedback sessions, we will have a lot of advantages. Some advantages are summarized as follows:

1) *Feedback sessions can be considered as a process of resampling*. If we view the original URLs in the search results as original samples, then feedback sessions can be viewed as the "processed" or "resampled" samples which differ from the original samples and reflect user information needs. Without resampling, there could be many noisy URLs in the search results, which are seldom clicked by users. If we cluster the search results with these noisy ones, the performance of clustering will degrade greatly. However, feedback sessions actually "resample" the URLs and exclude those noisy ones. Therefore, our method is much better than Method I. Furthermore, the resampling

TABLE 2
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

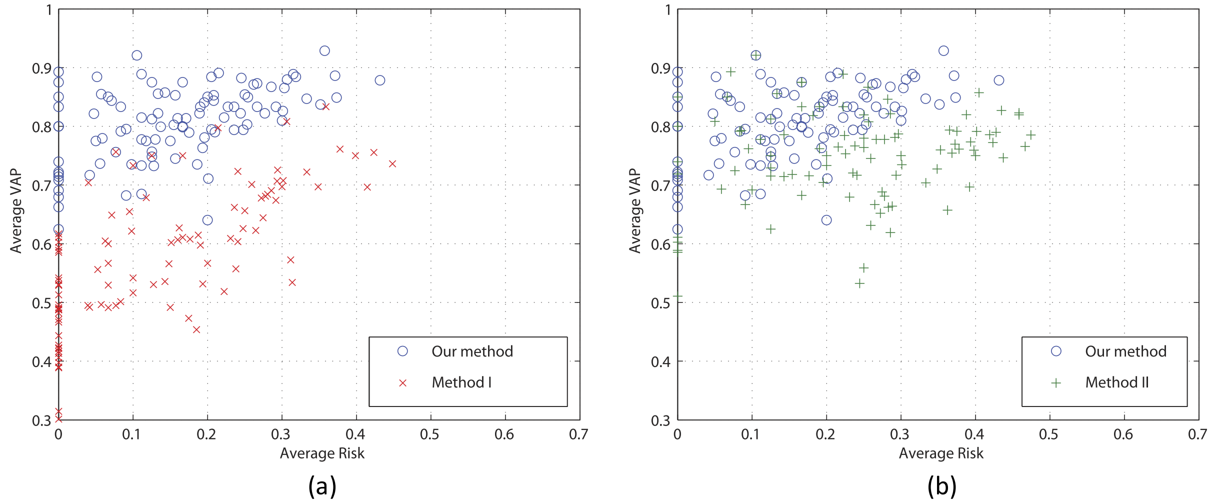


Fig. 9. Comparison of three methods for 100 most ambiguous queries. Each point represents the average $Risk$ and VAP of a query when evaluating the performance of restructuring the search results.

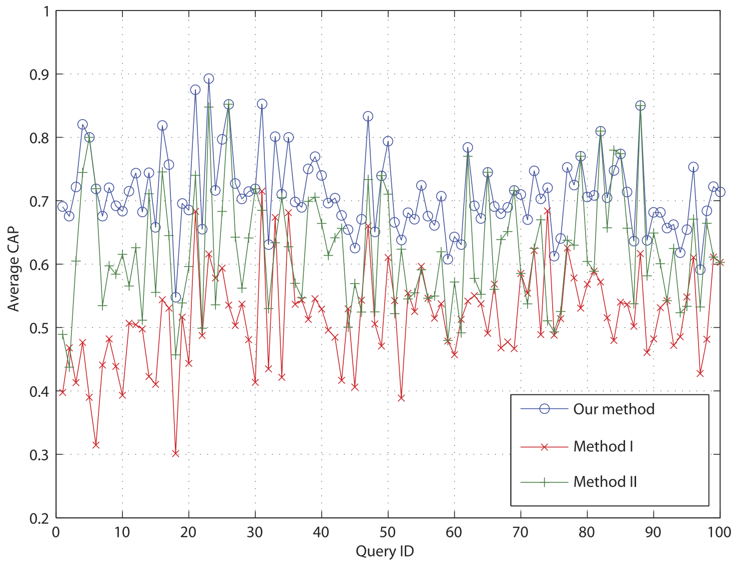


Fig. 10. The chart of CAP comparison of three methods for 100 most ambiguous queries.

TABLE 3
CAP Comparison of Three Methods for 100 Most Ambiguous Queries

Method	Mean Average VAP	Mean Average Risk	Mean Average CAP
Our method	0.807	0.159	0.715
Method I	0.583	0.138	0.525
Method II	0.750	0.231	0.624

by feedback sessions brings the information of user goal distribution to the new samples. For instance, most URLs in the search results of the query “the sun” are about the sun in nature while most feedback sessions are about the newspaper. Therefore, the introduction of feedback sessions provides a more reasonable way for clustering.

2) *Feedback session is also a meaningful combination of several URLs*. Therefore, it can reflect user information need more precisely and there are plenty of feedback sessions to be analyzed. For example, in Fig. 11, the solid points represent the clicked URLs mapped into a 2D space and we suppose that users have two search goals: the star points belong to one goal and the circle points belong to the other goal. The large ellipse in Fig. 11b represents a feedback session which is the combination of several clicked URLs. (In order to clarify the problem, we consider that feedback sessions only consist of click URLs here. However, if unclicked URLs are taken into account to construct feedback sessions, they will contain more information and be more efficient to be clustered.) Since the number of the different clicked URLs may be small, if we perform clustering directly on the points, it is very difficult to segment them precisely, as shown in Fig. 11a. However, supposing that most users have only one search goal, it is much easier to segment the

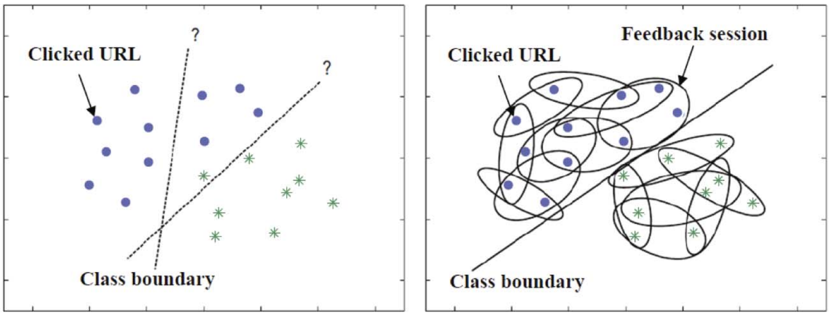


Fig. 11. Clustering of the clicked URLs and feedback sessions, where points represent clicked URLs, ellipses represent feedback sessions, and lines represent class boundaries. Supposing that the clicked URLs have two classes, the points in the left figure are hard to be segmented directly, while the ellipses in the right figure are easier to be segmented.

ellipses in Fig. 11b. From another point of view, feedback sessions can also be viewed as a preclustering of the clicked URLs for a more efficient clustering. Moreover, the number of the combinations of the clicked URLs can be much larger than the one of the clicked URLs themselves. Therefore, our method is better than Method II.

7 Related Work

In recent years, many works have been done to infer the so-called user goals or intents of a query [13], [14], [17]. But in fact, their works belong to query classification. Some works analyze the search results returned by the search engine directly to exploit different query aspects [6], [20]. However, query aspects without user feedback have limitations to improve search engine relevance. Some works take user feedback into account and analyze the different clicked URLs of a query in user click-through logs directly, nevertheless the number of different clicked URLs of a query may be not big enough to get ideal results. Wang and Zhai clustered queries and learned aspects of these similar queries [18], which solves the problem in part. However, their method does not work if we try to discover user search goals of one single query in the query cluster rather than a cluster of similar queries. For example, in [18], the query "car" is clustered with some other queries, such as "car rental," "used car," "car crash," and "car audio." Thus, the different aspects of the query "car" are able to be learned through their method. However, the query "used car" in the cluster can also have different aspects, which are difficult to be learned by their method. Some other works introduce search goals and missions to detect session boundary hierarchically [11]. However, their method only identifies whether a pair of queries belong to the same goal or mission and does not care what the goal is in detail.

A prior utilization of user click-through logs is to obtain user implicit feedback to enlarge training data when learning ranking functions in information retrieval. Thorsten Joachims did many works on how to use implicit feedback to improve the retrieval quality [8], [9], [10]. In our work, we consider feedback sessions as user implicit feedback and propose a novel optimization method to combine both clicked and unclicked URLs in feedback sessions to find out what users really require and what they do not care.

One application of user search goals is restructuring web search results. There are also some related works focusing on organizing the search results [6], [18], [20]. In this paper, we infer user search goals from user click-through logs and restructure the search results according to the inferred user search goals.

8 Conclusion

In this paper, a novel approach has been proposed to infer user search goals for a query by clustering its feedback sessions represented by pseudo-documents. First, we introduce feedback sessions to be analyzed to infer user search goals rather than search results or clicked URLs. Both the clicked URLs and the unclicked 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, we map 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. Experimental results on user click-through logs from a commercial search engine demonstrate the effectiveness of our proposed methods.

The complexity of our approach is low and our approach can be used in reality easily. For each query, the running time depends on the number of feedback sessions. However, the dimension of $\mathbf{F}_{fs}$ in (3) and (5) 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.

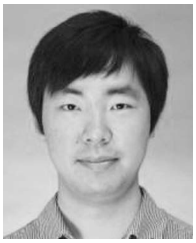
Acknowledgments

The authors would like to thank the support of NSFC (61025005, 61129001, 61221001), 973 Program (2010CB731401, 2010CB731406), the 111 Project (B07022), the Open Project Program of the National Library of Pattern Recognition (NLPR), and Shanghai Pujiang Program (12PJ1404300).

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 Contextual 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 Clickthrough Data," *Text Mining,* J. Franke, G. Nakhaeizadeh, and I. Renz, eds., pp. 79-96, Physica/Springer Verlag, 2003.
[9] 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.
[10] 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.
[11] 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.
[12] 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.
[13] 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.
[14] 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.
[15] 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.
[16] 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.
[17] 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.
[18] 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.
[19] J.-R Wen, J.-Y Nie, and H.-J Zhang, "Clustering User Queries of a Search Engine," *Proc. Tenth Int'l Conf. World Wide Web (WWW '01),* pp. 162-168, 2001.
[20] 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.



Zheng Lu received the BS degree from Fudan University, Shanghai, China, in 2005, and the ME degree from Shanghai Jiao Tong University, China, in 2008, both in electrical engineering. He is currently working toward the PhD degree at the Institute of Image Communication and Information Processing, Shanghai Jiao Tong University, China. His current research interests include information retrieval, machine learning and video processing. He is a student member of the IEEE.

Hongyuan Zha received the BS degree in mathematics from Fudan University in Shanghai in 1984, and the PhD degree in scientific computing from Stanford University in 1993. He was a faculty member of the Department of Computer Science and Engineering at the Pennsylvania State University from 1992 to 2006, and he worked from 1999 to 2001 at Inktomi Corporation. His current research interests include computational mathematics, machine learning applications and information retrieval.

Xiaokang Yang received the BS degree from Xiamen University, China, in 1994, the ME degree from the Chinese Academy of Sciences, Beijing, China, in 1997, and the PhD degree from Shanghai Jiao Tong University, China, in 2000. He is currently a professor with the Institute of Image Communication and Information Processing, Department of Electronic Engineering, Shanghai Jiao Tong University, China. From April 2002 to October 2004, he was a research scientist in the Institute for Infocomm Research, Singapore. His current research interests include scalable video coding, video transmission over networks, video quality assessment, digital television, and pattern recognition. He is a senior member of the IEEE.

Weiyao Lin received the BE and ME degrees from Shanghai Jiao Tong University, China, in 2003 and 2005, respectively, and the PhD degree from the University of Washington, Seattle, in 2010, all in electrical engineering. Since 2010, he has been an assistant professor in the Institute of Image Communication and Information Processing, Department of Electronic Engineering, Shanghai Jiao Tong University. His current research interests include video processing, machine learning, computer vision, video coding, and compression. He is a member of the IEEE.

Zhaohui Zheng received the bachelor's degree from Tsinghua University, the master's degree from the Institute of Automation at Chinese Academy of Sciences, and the PhD degrees from the State University of New York at Buffalo. He is currently the head of Yahoo! Labs Beijing charged with both scientific improvements for Yahoo!s existing core products and science-driven product innovations in the areas of search, advertising, media and mobile. Prior to his current role, he was a science director responsible for relevance sciences for Yahoo! search and audience products, including ranking science for web and vertical search worldwide, and recommendation science for Yahoo! properties and content networks. He has broad experience and expertise, having coauthored numerous papers in the areas of search and advertising sciences, machine learning, information retrieval, natural language processing, and data mining including the best student paper in ACM SIGIR 2011.