Sampling Based Estimation of Terrestrial Biodiversity through Extrapolation Crowdsource

L.Vishnu Priya[1], M. Barathi[2]

[1]L.Vishnu Priya,M.E/CSE ,Ganadipathy Tulsi's Jain Engineering College,Vellore,India-632102

vishnulakshmipathy@gmail.com

[2] M.Barathi,HOD/CSE,Ganadipathy Tulsi's Jain Engineering College,Vellore,India-632102

bharathi.damu@gmail.com

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. Web Service Composition (WSC) is the process of connecting multiple webservices to create higher-level business processes. The main aim of WSC is automatically select, integrate and call the multiple web services to achieve a user-defined objective where it executes workflows supplying complex user needs. Upon doing that, the QoS is considered to be the main concern as, a cross-domain request is likely potential. To enhance the QoS, concept of HTTP Redirect is applied in QoS-aware service composition used for web service repository. This can be achieved by CORS (Cross Origin Resource Sharing) method where the JSONP (Java Script Object Notation with Padding) technique is internally used to make a cross domain request.

Keywords— Crowdsource DB, searching technique, Fuzzy clustering.

1. INTRODUCTION

Accurately measuring the semantic similarity between words is a important problem in web mining, information retrieval, and natural language processing. Web mining applications such as, community extraction, relation detection, and entity disambiguation, require the ability to accurately measure the semantic similarity between concepts or entities. Efficient estimation of semantic similarity between words is critical for various natural language processing tasks such as word sense disambiguation (WSD), textual entailment, and automatic text summarization. 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[7].Analysing 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. Since user feedback is not considered, many noisy search results that are not clicked by any users may be analysed as well which consists of snippets, page-counts and support vector machine[6],[2]. Therefore, this kind of methods cannot infer user search goals precisely. For example, apple is frequently associated with computers on the web. However, this sense of apple is not listed in most general-purpose thesauri or dictionaries.

USER AUTHENTICATION:

1.1.Login

In computer security, a login or logon is the process by which individual access to a computer system is controlled by identifying and authenticating the user referring to credentials presented by the user. A user can log in to a system to obtain access and can then log out or log off when the access is no longer needed. To log out is to close off one's access to a computer system after having previously logged in.

1.2.User Search Log

The user enters the queries to the search engine. The queries are maintained as a log and the results will be produced based on the keywords. The search goals for a query and depicting each goal with some keywords automatically[2],[3] .The user's queries are saved.

1.3. Feedback Sessions

The feedback sessions is defined as the series of both clicked and un clicked URLs and ends with the last URL that was clicked in a session from user click-through logs. Then we map the feedback sessions to pseudo-documents which can

effectively reflect user information needs[8],[4]. We combine the enriched URL's in a feedback sessions to form a pseudo document. The feedback session is based on a single session .and also it can be extended to the whole session. So besides the clicked URLs, the un clicked ones before the last click should be a part of the user feedbacks[3] .For inferring user search goals it is more efficient to analyze the feedback sessions than to analyze the search results or clicked URLs directly.

1.4.Pseudo Documents

The feedback sessions vary a lot for different clicks through and queries ,it is not suitable to directly use the feedback sessions some method id needed to represent the feedbacks in a more efficient way. The search log will be represented as 0 in the click sequence. The binary vector is used to represent the feedback sessions 1 as clicked and 0 as un clicked.Steps to build pseudo documents[4]. Each URL is represented as a small text paragraph then some textual process is implemented as text paragraphs such as transforming all the letters to lower case, stemming and removing stop words.Forming pseudo documents based on URL representations:Process of combining both clicked and un clicked URL's in the feedback sessions

1.5.Clustering the Pseudo Documents

The Pseudo documents are clustered into K means clustering .It performs clustering based on the five values[6]. The terms with the highest values in the centre points are used as the keywords to depict user search goals. Similar queries may not share query-terms but they do share terms in the documents selected by the users. Thus we avoids the problems of comparing and clustering sparse collection of vectors in which similar queries are difficult to find a problem that appears in previous works on clustering .So we do rank the suggested queries based on two criteria's:

1.The similarity of the queries to input query (the query submitted to the search engine)

2.The support which measures how much the answers of the query have attracted the user's attention.

1.6. Final Restructured Results

The results are restructured based on the evaluation of web search goals. This approach Is called CAP(Classified Average Precision).Search engines will returns millions of search results so. It is necessary to organize them to make it easier for users to find what they want. The user search goals are represented as the vectors .

So we perform categorization by choosing the smallest distance between the URL vector and user-search – goal vectors. By this way the results can be restructured according to the inferred user search goals.

2. OVERVIEW OF EXISTING SYSTEMS

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[3],[8]. 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.

The drawbacks of the existing system is What users care about varies a lot for different queries, finding suitable predefined search goal classes is very difficult and impractical.Analysing the clicked URLs directly from user click-through logs to organize search results[9],[12].However, this method has limitations since the number of different clicked URLs of a query may be small. Since user feedback is not considered, many noisy search results that are not clicked by any users may be analysed as well. Therefore, this kind of methods cannot infer user search goals precisely.Only identifies whether a pair of queries belongs to the same goal or mission and does not care what the goal is in detail.

3. PROPOSED APPROACH

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. Then, we propose a novel optimization method to map feedback sessions to pseudo-documents which can efficiently reflect user information needs[10]. At last, we cluster these pseudo documents to infer user search goals and depict them with some keywords.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 we propose this novel criterion "Classified Average Precision" to evaluate the restructure results[2],[7]. Based on the proposed criterion, we also describe the method to select the best cluster number. We first propose a novel approach to infer user search goals for a query by clustering our proposed feedback sessions. Then, we propose a novel optimization method to map feedback sessions to pseudo-documents which can efficiently reflect user information needs.

The advantages of proposed system are 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[2],[7],[12].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 go inference based on restructuring web search results.Thus, we can determine the numeral of user search goals for a query.

4. DESIGN

The authors proposed an approach to compute the semantics similarity between words or entities using text snippets. But in this project we are going to implement and compute the semantic similarity between words in Search engine without using Snippets or Support Vector Machines. Because using Snippets or Support Vector Machines makes the job of finding similarity easier.Therefore, this kind of methods cannot infer user search goals precisely. However, this sense of apple is not listed in most general-purpose dictionaries.

Efficient estimation of semantic similarity between words is critical for various natural language processing tasks such as word sense disambiguation (WSD), textual entailment, and automatic text summarization. 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.

An information retrieval, one of the main problems is to retrieve a set of documents that is semantically related to a given user query[8],[10]. Information need is a user's particular desire to obtain information to satisfy his/her need.Since user feedback is inot considered, many noisy search results that are not clicked by any users may be analysed as well which consists of snippets, page-counts and support vector machine[4],[9].

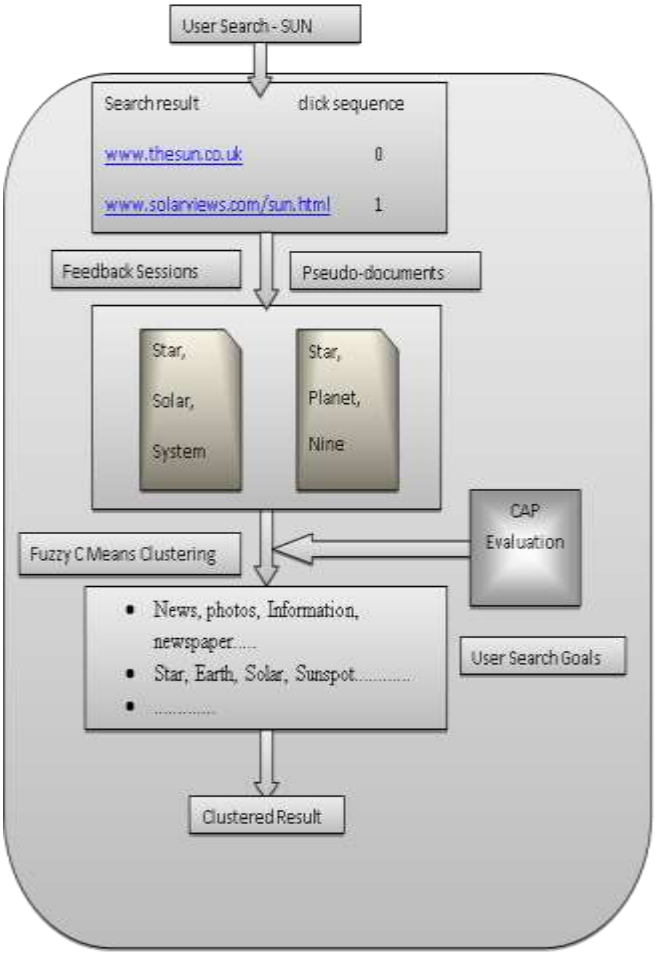

Fig 1.Framework of our approach

5. EXPERIMENT

5.1. Ambiguous Query

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[1],[2].

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. Queries are submitted to the search engine represents the information needs to user.

Fig 2.Ambiguous query

5.2. Restructure web search results

We need to restructure web search results according to user search goals by grouping the search results with the same search goal users with different search goals can easily find what they want. User search goals represented by some keywords can be utilized in query recommendation.The distributions of user search goals can also be useful in applications such as re-ranking web search results that contain different user search goals[3],[6].Due to its usefulness, many works about user search goals analysis have been investigated.

Retrieving accurate information for users in Search Engine faces a lot of problems.This is due to accurately measuring the semantic similarity between words is an important problem. For example, the word "apple" consists of two meaning one indicates the fruit apple and the other is the apple company. So retrieving accurate information to users to such kind of similar words is challenging.They can be summarized into three classes: query classification, search result reorganization, and session boundary detection[1],[2].

Fig 3.Registration

5.3. Feedback Sessions

The 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 un clicked ones before the last click should be a part of the user feedbacks [5],[7],[8].Feedback sessions of query are extracted from user click through logs and mapped pseudo documents and depicted with some keywords. Feedback session consists of both clicked and unclicked URLs and end with the last URL was clicked in a single session.

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[3],[9]. Therefore, for inferring user search goals, it is more efficient to analyze the feedback sessions than to analyze the search results or clicked URLsdirectly.

Fig 4.Feedback session

5.4. Pseudo document

In this paper, we need to map feedback session to pseudo documents User Search goals. The building of a pseudo-document includes two steps. One is representing the URLs in the feedback session[3],[4],[17].Then, some textual processes are implemented to those text paragraphs, such as transforming all the letters to lowercases, stemming and removing stop words. Another one is 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. Therefore, besides the clicked URLs, the un clicked ones before the last click should be a part of the user feedbacks[5].

Moreover, the distributions of different user search goals can be obtained conveniently after feedback sessions are clustered[3].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.

Fig 5..Keyword Maintenance

5.5. User Search Goals

We cluster pseudo-documents by FCM clustering which is simple and effective. Since we do not know the exact number of user search goals for each query, we set number of clusters to be five different values and perform clustering based on these five values, respectively[3],[4].

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.

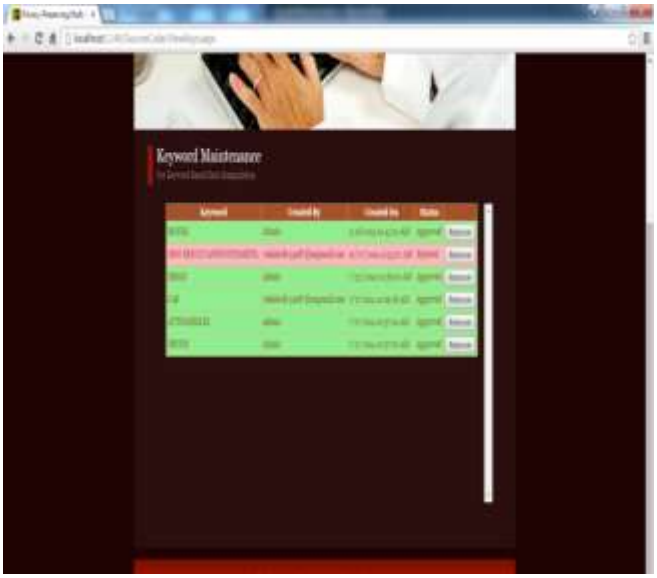

Fig 6.View Keywords

The center point of a cluster is computed as the average of the vectors of all the pseudo-documents in the cluster[6],[7]. Words that are similar to each other are grouped into the same cluster. Each cluster is characterized by a membership function with statistical mean and deviation.

5.6 .Fuzzy c-means clustering

Feature clustering is a powerful method to reduce the dimensionality of feature vectors for text classification. In this paper, we propose a fuzzy similarity-based self-constructing algorithm for feature clustering.

The words in the feature vector of a document set are grouped into clusters, based on similarity test. Words that are similar to each other are grouped into the same cluster. Each cluster is characterized by a membership function with statistical mean and deviation. When all the words have been fed in, a desired number of clusters are formed automatically. We then have one extracted feature for each cluster. The extracted feature, corresponding to a cluster, is a weighted combination of the words contained in the cluster.

By this algorithm, the derived membership functions match closely with and describe properly the real distribution of the training data. Besides, the user need not specify the number of extracted features in advance, and trial-and-error for determining the appropriate number of extracted features can then be avoided. Experimental results show that our method can run faster and obtain better extracted features than other methods. Fuzzy clustering is a class of algorithms for cluster analysis in which the allocation of data points to clusters is not "hard" (all-or-nothing) but "fuzzy" in the same sense as fuzzy logic.

Data clustering is the process of dividing data elements into classes or clusters so that items in the same class are as similar as possible, and items in different classes are as dissimilar as possible. Depending on the nature of the data and the purpose for which clustering is being used, different measures of similarity may be used to place items into classes, where the similarity measure controls how the clusters are formed. Some examples of measures that can be used as in clustering include distance, connectivity, and intensity.

In hard clustering, data is divided into distinct clusters, where each data element belongs to exactly one cluster. In fuzzy clustering (also referred to as soft clustering), data elements can belong to more than one cluster, and associated with each element is a set of membership levels.

These indicate the strength of the association between that data element and a particular cluster. Fuzzy clustering is a process of assigning these membership levels, and then using them to assign data elements to one or more clusters. One of the most widely used fuzzy clustering algorithms is the Fuzzy C-Means (FCM) Algorithm. The FCM algorithm attempts to partition a finite collection of n elements into a collection of c fuzzy clusters with respect to some given criterion.

Given a finite set of data, the algorithm returns a list of c cluster centres and a partition matrix, where each element wij tells the degree to which element xi belongs to cluster cj. Like the k-means algorithm, the FCM aims to minimize an objective function. The standard function is:

$$w_k(x) = \frac{1}{\sum_j \left(\frac{d(\text{center}_k, x)}{d(\text{center}_j, x)}\right)^{2/(m-1)}}.$$

which differs from the k-means objective function by the addition of the membership values uij and the fuzzifier m. The fuzzifier m determines the level of cluster fuzziness. A large m results in smaller memberships wij and hence, fuzzier clusters. In the limit m = 1, the memberships wij converge to 0 or 1, which implies a crisp partitioning.

In the absence of experimentation or domain knowledge, m is commonly set to 2. The basic FCM Algorithm, given n data points (x1, . . .,xn) to be clustered, a number of c clusters with (c1, . ., cc) the center of the clusters, and m the level of cluster fuzziness. In fuzzy clustering, every point has a degree of belonging to clusters, as in fuzzy logic, rather than belonging completely to just one cluster. 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 wk(x). With fuzzy c-means, the centroid of a cluster is the mean of all points, weighted by their degree of belonging to the cluster.The degree of belonging, wk(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 c-means algorithm is very similar to the k-means algorithm:

- Choose a number of clusters.
- Assign randomly to each point coefficients for being in the clusters.
- Repeat until the algorithm has converged (that is, the coefficients' change between two iterations is no more than, the given sensitivity threshold):
- Compute the centroid for each cluster, using the formula above.

For each point, compute its coefficients of being in the clusters, using the formula above.

The algorithm minimizes intra-cluster variance as well, but has the same problems as k-means; the minimum is a local minimum, and the results depend on the initial choice of weights.Using a mixture of Gaussians along with the expectation-maximization algorithm is a more statistically formalized method which includes some of these ideas: partial membership in classes .Another algorithm closely related to Fuzzy C-Means is Soft K-means.Fuzzy c-means has been a very important tool for image processing in clustering objects in an image. In the 70's, mathematicians introduced the spatial term into the FCM algorithm to improve the accuracy of clustering under noise.

6.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[7],[8]. 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, we map feedback sessions to pseudo documents to approximate goal texts in user minds[7],[12]. 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.

7. FUTURE WORK

In this system, for the first time we define and solve the problem of multi-keyword ranked search over encrypted cloud data, and establish a variety of privacy requirements[7]. Among various multi-keyword semantics, we choose the efficient similarity measure of "coordinate matching", i.e., as many matches as possible, to effectively capture the relevance of outsourced documents to the query keywords, and use "inner product similarity" to quantitatively evaluate such similarity measure[9],[11].

For meeting the challenge of supporting multi-keyword semantic without privacy breaches, we propose a basic idea of MRSE using secure inner product computation. Then we give two improved MRSE schemes to achieve various stringent privacy requirements in two different threat models[3]. Thorough analysis investigating privacy and efficiency guarantees of proposed schemes is given, and experiments on the real-world dataset show our proposed schemes introduce low overhead on both computation and communication. In our future work, we will explore supporting other multi-keyword semantics (e.g., weighted query) over encrypted data and checking the integrity of the rank order in the search result.

REFERENCES

1.H. Chen and S. Dumais, "Bringing Order to the Web: Automatically Categorizing Search results", Proc. SIGCHI Conf. Human Factors in ComputinSystems(SIGCHI'00), PP.145-152, 2000.

2.C. K Huang, L.F Chien and Y. J Ovang, "Relevant Term Suggestion in Interactive Web search basd on Contextual Information in Query Session Logs", J.Am.Soc. for information Science and Technology, Vol. 54, no. 7, pp. 638-649, 2003.

3.T. Joachnims, "Evaluating Retrieval Performance Using Clickthrough Data", Text Mining, J. Franke, G.Nakhaeizadeh and I. Renz, edu, pp.79-96. Physiscs/Springer Verlag 2003.

4.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.

5.R. Jones and K.L. Klinkner, "Beyond the Session Timeout: Automatic Segmentation of Search Topics in Query Logs," Proc. 17th ACM Conf. in univer. Information and Knowledge Management (CIKM '08), pp. 699-708, 2008.

6.R. Jones, B. Rey, O. Madani, and W. Greiner, "Generating inQuery Substitutions," Proc. 15th Int'l Conf. World Wide Web (WWW '06), pp. 387-396, 2006.

7.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.

8.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.

9.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

10.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.

11.T. Joachnims, "Evaluating Retrieval Performance Using Clickthrough Data", Text Mining, J. Franke, G.Nakhaeizadeh and I. Renz, edu, pp.79-96. Physiscs/Springer Verlag 2003.

12.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.

13.R. Jones and K.L. Klinkner, "Beyond the Session Timeout: Automatic Segmentation of Search Topics in Query Logs," Proc. 17th ACM Conf. in univer. Information and Knowledge Management (CIKM '08), pp. 699-708, 2008.

14.R. Jones, B. Rey, O. Madani, and W. Greiner, "Generating inQuery Substitutions," Proc. 15th Int'l Conf. World Wide Web (WWW '06), pp. 387-396, 2006.

15.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.

16.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.

17.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

18.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.

