Mining Search Query Logs for Spoken Language Understanding

Dilek Hakkani-Tür, Gokhan Tür, Asli Celikyilmaz
Microsoft, Mountain View, CA 94041, USA
dilek|gokhan.tur|asli@ieee.org

Abstract

In a spoken dialog system that can handle natural conversation between a human and a machine, spoken language understanding (SLU) is a crucial component aiming at capturing the key semantic components of utterances. Building a robust SLU system is a challenging task due to variability in the usage of language, need for labeled data, and requirements to expand to new domains (*movies, travel, finance*, etc.). In this paper, we survey recent research on bootstrapping or improving SLU systems by using information mined or extracted from web search query logs, which include (natural language) queries entered by users as well as the links (web sites) they click on. We focus on learning methods that help unveiling hidden information in search query logs via implicit crowd-sourcing.

1 Introduction

Building a robust spoken dialog system involves human language technologies to cooperate to answer natural language (NL) user requests. First user's speech is recognized using an automatic speech recognition (ASR) engine. Then a spoken language understanding (SLU) engine extracts their meaning to be sent to dialog manager for taking the appropriate system action.

Three key tasks of an SLU system are domain classification, intent determination and slot filling (Tur and Mori, 2011). While the state-of-the-art SLU systems rely on data-driven methods, collecting and annotating naturally spoken utterances to train the required statistical models is often costly and time-consuming, representing a significant barrier to deployment. However, previous work shows that it may be possible to alleviate this hurdle by leveraging the abundance of implicitly labeled web search queries in search engines. Large-scale engines, e.g., Bing or Google, log more than 100M queries every day. Each logged query has an associated set of URLs that were clicked after the users entered the query. This information can be valuable for building more robust SLU components, therefore, provide (noisy) supervision in training SLU models. Take domain detection problem: Two users who enter different queries but click on the same URL (www.hotels.com) would probably be searching for concepts in the same domain ("hotels" in this case).

The use of click information obtained through massive search query click logs has been the focus of previous research. Specifically, query logs have been used for building more robust web search and better information retrieval (Pantel and Fuxman, 2011; Li et al., 2008), improve personalization experience and understand social networking behaviors (Wang et al., 2011), etc. The use of query logs in spoken dialog research is fairly new. In this paper, we will survey the recent research on utilizing the search query logs to obtain more accurate and robust spoken dialog systems, focusing on the SLU. Later in the discussion section, we will discuss the implimications on the dialog models.

The paper is organized as follows: In § 2, we briefly describe query click logs. We then summarize recent research papers to give a snapshot of how user search queries are being used in § 3, and how information from click-through graphs (queries and

NAACL-HLT 2012 Workshop on Future directions and needs in the Spoken Dialog Community: Tools and Data, pages 37–40,
Montréal, Canada, June 7, 2012.

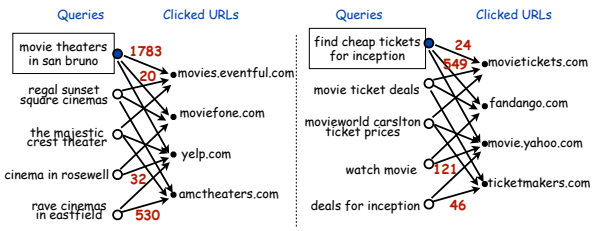

Figure 1: A sample query click graph. The squared queries are samples from training data which are natural language utterances. Edges include click frequencies from query to link.

clicked links) are exploited to boost the SLU performance. Lastly, we discuss possible future directions.

2 What are Query Click Logs (QCL)?

QCL are logs of unstructured text including both the users queries sent to a search engine and the links that the users clicked on from the list of sites returned by that search engine. A common representation of such data is a bi-partite query-click graph as shown in (Fig 1), where one set of nodes represents queries, and the other set of nodes represents URLs, and an edge is placed between two nodes representing a query q and a URL u, if at least one user who typed the q clicked on u.

Traditionally, the edge of the click graph is weighted based on the raw click frequency (number of clicks) from a query to a URL. Some of the challenges in extracting useful information from QCL is that the feature space is high dimensional (there are thousands of url clicks linked to many queries), and there are millions of queries logged daily.

3 Exploiting NL Search Queries for SLU

Previous work on web search has benefited from the use of query click logs for improving query intent classification. Li *et al.* use query click logs to determine the domain of a query (typically keyword search queries), and then infer the class memberships of unlabeled queries from those of the labeled search queries using the URLs the users clicked (Li et al., 2009; Li et al., 2008). QCL have been used to extract named-entities to improve web search and ad publishing experience (Hillard and Leggetter, 2010) using (un)supervised learning methods on keyword based search queries. Different from previous research, in this paper we focus on recent research that utilize NL search queries to boost the performance of SLU components, i.e., domain detection, intent determination, and slot filling.

In (Hakkani-Tur et al., 2011a), they use the search query logs for *domain classification* by integrating noisy supervision into the semi-supervised label propagation algorithm, and sample high-quality query click data. Specifically, they extract a set of queries, whose users clicked on the URLs that are related to their target domain categories. Then they mine query click logs to get all instances of these search queries and the set of links that were clicked on by search engine users who entered the same query. They compare two semi-supervised learning methods, self-training and label propagation, to exploit the domain information obtained form the URLs user have clicked on. The analysis indicate that query sampling through semi-supervised learning enables extracting NL queries for use in domain detection. They also argue that using raw queries with and without the noisy labels in semi-supervised learning reduces domain detection error rate by 20% relative to supervised learning which uses only the manually labeled examples.

The search queries found in click logs and the NL spoken utterances are different in the sense that the search queries are usually short and keyword based compared to NL utterances that are longer and are usually grammatical sentences (see Fig. 1). Hence, in (Hakkani-Tur et al., 2012), they choose a statistical machine translation (SMT) approach to search query mining for SLU as sketched in Fig. 2. The assumption is that, users typically have conceptual intents underlying their requests when they interact with web search engine or use a virtual assistance system with built in SLU engine, e.g., "*avatar awards*" versus "*which awards did the movie avatar win?*". They translate NL queries into search queries and mine similar search queries in QCL. They also exploit QCL for bootstrapping domain detection models, using only the NL queries hitting to seed domain indicator URLs (Hakkani-Tur et al., 2011c). Specifically, if one needs to detect a domain detector for the hotels domain, the queries hitting hotels.com, or tripadvisor.com, may be used to mine.

Query click logs have been explored for *slot filling* models as well. The slot filling models of SLU

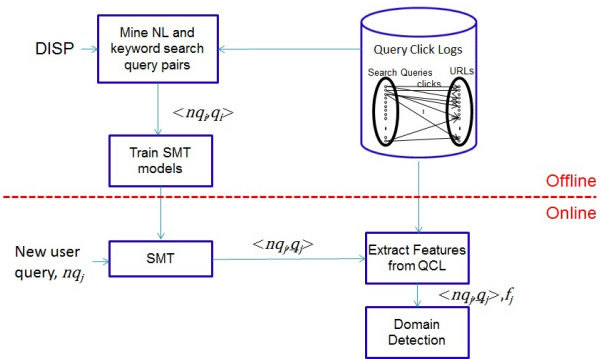

Figure 2: Using natural language to query language translation for mining query click logs.

aim to capture semantic components given the domain and a common way is to use gazetteer features (dictionaries specific to domain such as *movie-name* or *actors* in movie domain). In (Hillard et al., 2011), they propose to mine and weight gazetteer entries using query click logs. The gazetteer entries are scored using a function of posterior probabilities for that entry hitting a URL (compared to others URLs) and for that URL being related to the target domain. In such a schema the movie name "*gone with the wind*" gets higher score than the movie "*up*".

In (Tur et al., 2011), an unsupervised approach is presented to implicitly annotate the training data using the QCL. Being unsupervised, this method automatically populates gazetteers as opposed to manually crafted gazetteers. Specifically they use an abundant set of web search query logs with their click information (see Fig. 1). They start by detecting target URLs (such as `imdb.com/title` for the movie names). Then they obtain a list of entities and their target URLs (for example, www.imdb.com/title/tt047723 can be the target URL for the movie "*the count of monte carlo*". Then they extract all queries hitting those links if they include that entity. This method enables automatically obtaining annotated queries such as: "*review of* **the hand**" or "**mad men** *season one synopsis*" (***bold*** terms are automatically discovered entities.)

4 Mining Click Graph Features for SLU

In the previous section, we presented examples of recent research that use queries obtained from QCL to bootstrap and improve SLU models. Note that each query in QCL is linked to one or many web sites (links), which indicate a certain feature of the query (queries that the *hotels.com* linked are clicked after they are entered might indicate hotels domain). Such features extracted from QCL data (called click-through features) has been demonstrated to significantly improve the performance of ranking models for Web search applications (Gao et al., 2009), estimating relations between entities and web search queries (Pantel and Fuxman, 2011), etc.

In SLU research community, only recently the use of click-through features has shown to improve the performance of domain and intent of NL user utterances. In one study (Hakkani-Tur et al., 2011b), instead of mining more data to train a domain classifier with lexical features, they enrich their features using the click-through features with the intuition that the queries with similar click patterns should be semantically similar. They search all the NL utterances in the training data set amongst the search queries. Once they obtain search queries, they pull the list of clicked URLs and their frequencies for each query which represent the click features. To reduce the number of features, they extract only the base URLs (such as opentable.com or wikipedia.com), as is commonly done in the web search literature. T use the list of the 1000 most frequently clicked base URLs for extracting classification features (QCL features). For each input user utterance, x_j, they compute $P(URL_i|x_j)$, where $i = 1..1000$. They compute the click probability distribution distance between a query and the queries in a target domain, D_k, using the KL divergence:

$$KL_k = KL(P(URL_i|x_j)||P(URL_i|D_k)) \quad (1)$$

Thus, for a given domain D_k, the KL_k and the domain with the lowest KL divergence are used as additional features.

Although the click-through are demonstrated to be beneficial for SLU models, such benefits, however, are severely limited by the data sparseness problem, i.e., many queries and documents have no or very few clicks. The SLU models thus cannot rely strongly on click-through features. In (Celikyilmaz et al., 2011), the sparsity issue of representing the queries with click-through features are investigated. They represent each unlabeled query from QCL as

a high dimensional sparse vector of click frequencies. Since the true dimensionality of a query is unknown (the number of clicks are infinitely many), they utilize an unbounded factor analysis approach and build an infinite dimensional latent factor analysis, namely the Indian Buffet Process (IBP) (Griffiths and Ghahramani, 2005), specifically to model the latent factor structure of the given set of queries. They implement a graph summarization algorithm to capture representative queries from a large set of unlabeled queries that are similar to a rather smaller set of labeled queries. They capture the latent factor structure of the labeled queries via IBP and reduce the dimensionality of the queries to manageable size and collect additional queries in this latent factor space. They use the new set of utterances boost the intent detection performance of SLU models.

5 Discussions and Future Directions

This paper surveyed previous research on the usage of the query click logs (the click through data) provide valuable statistics that can potentially improve performance of the SLU models. We presented several methods that has been used to extract information in the form of additional vocabulary, unlabeled utterances and hidden features to represent utterances. The current research is only the beginning, and most approaches such as query expansion, sentence compression, etc. can be easily adopted for dialog state update processes. Thus, the state-of-the art in NL understanding can be improved by:

• clustering of URLs as well as queries for extracting better features as well as to extend ontologies. The search community has access to vast amounts of search data that would benefit natural language processing research,

• mining multi-lingual data for transferring dialog systems from one language to others,

• mining information from search sessions, for example, users rephrasing of their own search queries for better results.

One issue that has been the topic of recent discussions is the accessibility of QCL data to researchers. Note that, QCL is not a crowd-source data that only large web search organizations like Google or Microsoft Bing can mine and exploit for NL understanding, but various other forms may be implemented by interested researchers by using a simple web service or a mobile app (such as AT&T SpeakIt or Dragon Go) or using a targeted search engine.

References

A. Celikyilmaz, D. Hakkani-Tur, and G. Tur. 2011. Leveraging web query logs to learn user intent via bayesian latent variable model. In *ICML'11 - WS on Combining Learning Strategies to Reduce Label Cost*.

J. Gao, J.-Y. Nie, W. Yuan, X. Li, and K. Deng. 2009. Smoothing clickthrough data for web search ranking. In *SIGIR'09*.

T. Griffiths and Z. Ghahramani. 2005. Infinite latent feature models and the indian buffet process. In *NIPS'05*.

D. Hakkani-Tur, G. Tur, and L. Heck. 2011a. Exploiting web search query click logs for utterance domain detection in spoken language understanding. In *ICASSP 2011*.

D. Hakkani-Tur, G. Tur, L. Heck, A. Celikyilmaz, A. Fidler, D. Hillard, R. Iyer, and S. Parthasarathy. 2011b. Employing web search query click logs for multi-domain spoken language understanding. In *ASRU'11*.

D. Hakkani-Tur, G. Tur, L. Heck, and E. Shriberg. 2011c. Bootstrapping domain detection using query click logs for new domains. In *Interspeech'11*.

D. Hakkani-Tur, G. Tur, R. Iyer, and L. Heck. 2012. Translating natural language utterances to search queries for slu domain detection using query click logs. In *ICASSP'12*.

D. Hillard and C. Leggetter. 2010. Clicked phrase document expansion for sponsored search ad retrieval. In *SIGIR'10*.

D. Hillard, A. Celikyilmaz, D. Hakkani-Tur, and G. Tur. 2011. Learning weighted entity lists from web click logs for slu. In *Interspeech'11*.

X. Li, Y.-Y. Wang, and A. Acero. 2008. Learning query intent from regularized click graphs. In *SIGIR08*.

X. Li, Y.-Y. Wang, and A. Acero. 2009. Extracting structured information from user queries with semi-supervised conditional random fields. In *ACM SIGIT'09*.

P. Pantel and A. Fuxman. 2011. Jigs and lures: Associating web queries with structured entities. In *ACL'11*.

G. Tur and R. De Mori, editors. 2011. *Spoken Language Understanding: Systems for Extracting Semantic Information from Speech*. John Wiley and Sons.

G. Tur, D. Hakkani-Tur, D. Hillard, and A. Celikyilmaz. 2011. Towards unsupervised spoken language understanding: Exploiting query click logs for slot filling. In *Interspeech'11*.

C. Wang, R. Raina, D. Fong, D. Zhou, J. Han, and G. Badros. 2011. Learning relevance from a heterogeneous social network and application in online targeting. In *SIGIR'11*.