The Wisdom of Advertisers: Mining Subgoals via Query Clustering

Takehiro Yamamoto[†‡♭*]
tyamamot@dl.kuis.kyoto-u.ac.jp

Tetsuya Sakai[†]
tetsuyasakai@acm.org

Mayu Iwata[†♯*]
iwata.mayu@ist.osaka-u.ac.jp

Chen Yu[†]
yu.chen@microsoft.com

Ji-Rong Wen[†]
jrwen@microsoft.com

Katsumi Tanaka[‡]
tanaka@dl.kuis.kyoto-u.ac.jp

†Microsoft Research Asia, China ‡Kyoto University, Japan ♭JST CREST, Japan ♯Osaka University, Japan

ABSTRACT

This paper tackles the problem of mining subgoals of a given search goal from data. For example, when a searcher wants to travel to London, she may need to accomplish several subtasks such as "book flights," "book a hotel," "find good restaurants" and "decide which sightseeing spots to visit." As another example, if a searcher wants to lose weight, there may exist several alternative solutions such as "do physical exercise," "take diet pills," and "control calorie intake." In this paper, we refer to such subtasks or solutions as subgoals, and propose to utilize sponsored search data for finding subgoals of a given query by means of query clustering. Advertisements (ads) reflect advertisers' tremendous efforts in trying to match a given query with implicit user needs. Moreover, ads are usually associated with a particular action or transaction. We therefore hypothesized that they are useful for subgoal mining. To our knowledge, our work is the first to use sponsored search data for this purpose. Our experimental results show that sponsored search data is a good resource for obtaining related queries and for identifying subgoals via query clustering. In particular, our method that combines ad impressions from sponsored search data and query co-occurrences from session data outperforms a state-of-the-art query clustering method that relies on document clicks rather than ad impressions in terms of purity, NMI, Rand Index, F_1-measure and subgoal recall.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval

General Terms

Algorithms, Experimentation

Keywords

User Intent, Sponsored Search, Query Clustering

1. INTRODUCTION

The information needs of a web search engine user are sometimes complex, and may span multiple queries or even multiple search sessions [21, 31]. When the user wants to travel to London, she may need to accomplish several subtasks such as "book flights," "book a hotel," "find good restaurants" and "decide which sightseeing spots to visit," and issue multiple queries accordingly, possibly over a length of time. As another example, for a user who wants to lose weight, there may exist several alternative solutions such as "do physical exercise," "take diet pills," and "control calorie intake." The user may not even be aware that these different solutions exist, so she will probably have to issue several queries to find out about them. In this paper, we refer to such subtasks or solutions as *subgoals*. Our precise definitions are as follows:

- A *search goal* is an *action* that the searcher wants to achieve, often represented by a verb plus possibly a noun phrase.
- A search goal *x* is a *subgoal* of another search goal *y* if achieving *x* helps the searcher to also achieve *y* either wholly or partially.

Note that while a subgoal "book flights" alone can only partially satisfy a search goal "travel (to) London," a subgoal "do physical exercise" may wholly satisfy "lose weight."

In this paper, we tackle the problem of automatically mining subgoals of a given search goal from data. To this end, we propose to utilize *sponsored search data* for finding subgoals of a given query by means of query clustering. Advertisements (ads) reflect advertisers' tremendous efforts in trying to match a given query with implicit user needs. Moreover, ads are usually associated with a particular action or transaction. We therefore hypothesized that they are useful for subgoal mining. To our knowledge, our work is the first to use sponsored search data for this purpose.

We further hypothesize that queries that represent a common subgoal are associated with similar ads, and employ a state-of-the-art query clustering algorithm [28] in order to mine subgoals

*This research was conducted while the first and third authors were interns at Microsoft Research Asia.

CIKM'12, October 29–November 2, 2012, Maui, HI, USA.

Table 1: Example search goals and their subgoals mined by our proposed method.

search goal: **lose weight**	
1.	fitness, gyms, health clubs, workout, ...
2.	alli, diet pills, best weight loss pills, hcg drops, ...
3.	diabetic recipes, diet recipes, healthy recipes, ...
4.	denise austin, jillian michaels, kathy smith
5.	high protein foods, protein, protein foods
6.	calorie counter
⋮	
search goal: **relieve stress**	
1.	cheap lexapro, generic lexapro, lexapro side effects, wellbutrin xl, wellbutrin xl 150 mg, ...
2.	baseball stress balls, stress ball, stress relief toy, stress toys
3.	body massage, massage therapist, massage therapy, stress factory, ...
4.	exercise heal, gaiam, holden, qigong
5.	anxiety medications, herbs anxiety
6.	zen garden
⋮	
search goal: **travel London**	
1.	cheap hotel london, london airport hotels, london hotel deals, london hotels, london luxury hotels, ...
2.	airfare london, airline tickets london, cheap airfare london, flights london, ...
3.	london 2012, london summer olympics, 2012 olympic games, 2012 olympics tickets, ...
4.	car rental london, london car rental
5.	london travel card, oyster card london
⋮	

from queries in the sponsored search data. Given a query that represents a search goal, our method first collects its related queries from sponsored search data, and then clusters them based on ad impressions as well as within-session query co-occurrences. Table 1 shows a few examples of goal-subgoal relationships mined using our proposed method. It can be observed that for the search goal "lose weight," query clusters that represent subgoals such as "do physical exercise," "take diet pills" and "control calorie intake" are obtained. Also, for the search goal "relieve stress," we can observe that query clusters that represent possible alternative solutions such as "take medicine," "buy stress relief toys" and "have a massage" are obtained. This hierarchy of search goals have many possible applications, including:

- Given a query, present the user with its possible subgoals, for example, as query suggestions;
- Given a query (e.g. "lose weight pills"), present the user with alternatives (e.g. "healthy diet recipes"), by identifying their common grand goal (e.g. "lose weight");
- Evaluating web search engines from the viewpoint of total user satisfaction, by means of goal-based evaluation as opposed to query-based evaluation as exemplified by nDCG [20]. Note that even though there are attempts at session-based IR evaluation [23], queries that represent subgoals of a search goal may span multiple sessions and users in general.

Figure 1 shows some example ads of a commercial search engine. In sponsored search, advertisers bid on various *bid phrases* so that their ads will be displayed in response to certain queries as

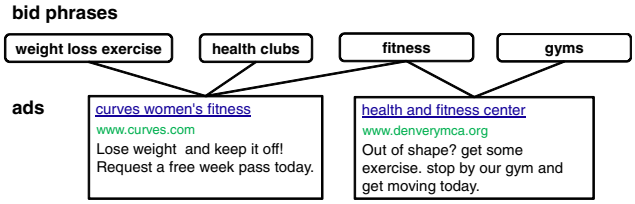


Figure 1: Example of ads and their bid phrases.

shown in the figure. Today's major commercial search engines have this kind of mechanism. According to our preliminary study with 10,000 head queries obtained over one recent week from a popular search engine, 68.3% of them had some ads displayed.

Our experimental results show that sponsored search data is a good resource for obtaining related queries and for identifying subgoals via query clustering. In particular, combining ad impressions with within-session query co-occurrence information outperforms a state-of-the-art query clustering method that uses document clicks rather than ad impressions, in terms of *purity, NMI, Rand Index,* F_1*-measure* and *subgoal recall* (See Section 4.4).

The rest of this paper is organized as follows. Section 2 discusses previous work related to our study. Section 3 describes our approach to mining subgoals using sponsored search data. Section 4 describes our experimental setup and Section 5 discusses the results. Finally, Section 6 concludes this paper.

2. RELATED WORK

2.1 Sponsored Search

Sponsored search has recently been an area of active research. The main research topics of this area have been the improvement of ads retrieval performance [6, 16, 27] and clickthrough rate (CTR) estimation for the retrieved ads [14, 32]. For example, Broder *et al.* [6] proposed a technique that leverages related organic (i.e. non-sponsored) search results as the source of query expansion to overcome the problem of low precision and recall of ads retrieval.

Some researchers have analyzed user behaviors in the context of sponsored search [7, 10, 17, 19]. In Jansen's experiments with e-commerce queries [17], the participants' relevance ratings for organic search results and those for ads were practically the same. Moreover, according to a CTR analysis by Danescu-Niculescu-Mizil *et al.* [10], users prefer ads that are dissimilar to organic results for navigational queries, and those that are similar to organic results for informational queries. The findings shown in both literature support the fact that ads play an important role in satisfying the users' information need along with organic search results.

2.2 Query Clustering

Next, we discuss prior art in query clustering, as our approach applies query clustering to sponsored search data in order to mine subgoals.

Query clustering is useful for understanding the underlying user intents and for improving query suggestions, and thus many query clustering techniques have been proposed. Most of existing work have relied on statistics derived from query session and clickthrough data: within-session query co-occurrences [13, 4, 29], similarity of clicked documents [2, 3, 30] and their combination [8, 28]. In this paper, we apply the query clustering algorithm proposed by Sadikov *et al.* [28] to sponsored search data, as it has been shown to achieve state-of-the-art performances through a large-scale user study. The main objective of this study is to show the usefulness of sponsored search data as a resource for mining subgoals: effective

query clustering methods other than that by Sadikov *et al.* are also applicable to this problem.

We hypothesized that using the session data alone is not sufficient for the purpose of mining subgoals, because queries issued for a single search goal may span multiple sessions over a length of time and even span multiple users. Note that sessions are defined based on 10-30 minutes of inactivity in many studies, and also that a single session may contain queries for several different search goals.

In contrast to session and clickthrough data, sponsored search data may have the following advantages for the purpose of subgoal mining: (a) we may be able to mine goal-subgoal relationships across sessions and across users; (b) since ads reflect the advertisers' tremendous effort in trying to match queries with the underlying user intents, we can leverage high-quality query-ad relationships that go beyond surface-level matching; (c) as ads are designed to make the user perform an action or transaction, they may directly reflect goals and subgoals.

2.3 Query Intent Categorization

Categorizing queries into predefined classes is an alternative to the aforementioned bottom-up clustering approaches for the purpose of understanding user intents. Many researchers [18, 22, 24] have tackled the problem of categorizing queries into *navigational*, *informational* and *transactional* [5]. Moreover, in the context of sponsored search, Dai *et al.* classified queries into *commercial* (e.g. buy or sell something) and *non-commercial* [9]. Guo and Agichtein [15] further refined the commercial category into *research* and *purchase*. However, these top-down approaches are not appropriate for our purpose, as we need to mine a variety of *unknown* subgoals for a given search goal.

2.4 Search Missions and Goals

The work that is most closely related to our present study is that by Jones and Klinkner [21], who introduced the concepts of *search mission* and *search goal*. According to their definitions, a search goal is an atomic information need represented by one or more queries, while a search mission is a set of related information needs represented by one or more search goals. Although our goal-subgoal relationships are also defined hierarchically just like their mission-goal relationships, ours are different from theirs: while their aim is to analyze individual search tasks of a searcher and thus their mission-goal relationships represent the hierarchical needs for the searcher, our aim is to mine more general hierarchical needs of searchers, rather than limiting ourselves to a session of a single searcher. In addition, although they proposed a model to automatically determine whether a given pair of queries in the same session shares the same search mission or goal, the candidate query pairs are a given. In contrast, given a query, our approach mines its subgoals from related queries in the sponsored search data.

More recently, Aiello *et al.* have proposed a clustering algorithm that clusters search missions into underlying *topics* [1]. Their aim is to find broad topical user profiles of search engine users from query logs, rather than finding mission-goal relationships. For example, they aim at finding the general *travel* intent from related missions such as "Find information on travel to London."

3. SUBGOAL MINING METHOD

3.1 Overview

In response to a query, commercial search engines often present sponsored search results in addition to organic search results. This happens when the query matches with the advertisers' bid phrases, and the associated ads are ranked according to their bids and estimated CTR [10]. The contents of each ad, devised by the advertisers, are typically like the ones shown in Figure 1. The sponsored search data used in our work leverages the above mechanism: some examples are shown in Table 2. Each data record is a triple, consisting of the query issued by a search engine user, the ad, and its impression count, which represents the number of times the ad was displayed in response to the query.

Table 2: Structure of sponsored search data in this study.

query	ad	impression
fitness	ad1	2,300
fitness	ad2	530
health clubs	ad1	1,880

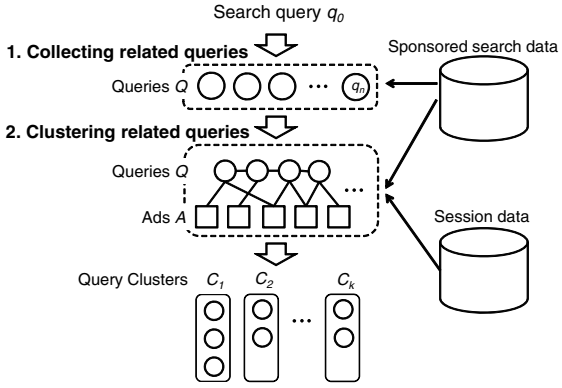


Figure 2: Overview of our method.

Figure 2 shows the overview of our subgoal mining method. In this paper, we refer to the query that is input to our method as *search query*, to distinguish it from its related queries that are to be clustered by our method. Given the search query q_0, our first step is to collect a set of related queries $Q = \{q_1, \ldots, q_n\}$ for a given n by utilizing the ad contents of the aforementioned sponsored search data. Then, our second step outputs a set of query clusters $\mathcal{C} = \{C_1, \ldots, C_k\}$ for a given k. This step uses both sponsored search data and session data and applies the clustering algorithm by Sadikov *et al.* [28].

Sections 3.2 and 3.3 describe the above two steps in detail.

3.2 Collecting Queries

Given the search query q_0 and the required number of related queries n, we first collect a set of related queries $Q = \{q_1, \ldots, q_n\}$ using the ad contents of the sponsored search data, as described below.

Let $\mathcal{A}$ be the entire set of ads archived in the sponsored search data, and let $\mathcal{A}_{q_0} (\subseteq \mathcal{A})$ be the set of all ads whose ad contents contain q_0. For example, note that the ad on the left in Figure 1 contains "lose weight" which could be a search query. For each query q in the sponsored search data such that $w_{\mathrm{imp}}(q, a) > 0$ for some $a \in \mathcal{A}_{q_0}$, where $w_{\mathrm{imp}}(q, a)$ denotes the impression count (See Table 2), we compute its total impression across the relevant ads: $\sum_{a \in \mathcal{A}_{q_0}} w_{\mathrm{imp}}(q, a)$. Then, we sort the queries by the total impression and take the top n. That is, we obtain n queries related to q_0, whose associated ads have high impressions.

3.3 Clustering Queries

Having thus obtained the set of related queries Q, we cluster the queries into k clusters, using the algorithm proposed by Sadikov *et al.* [28]. The original purpose of their clustering algorithm was to identify intents that are common across different query strings,

and it relied on two assumptions: (1) If two queries share the same intent, they are associated with the same clicked documents; and (2) If two queries share the same intent, they co-occur within the same session. For the purpose of mining subgoals from sponsored search data, we adapt the above assumptions as follows:

- If two queries represent the same subgoal intent, they are associated with the same ads.
- If two queries represent the same subgoal intent, they co-occur within the same session.

Thus, our departures from the original method by Sadikov *et al.* are: (a) We construct a *query-ad* graph instead of a *query-document* graph; and (b) We use *ad impressions* for computing the query-ad transition probability instead of *document clicks*. Below, we provide more details of the clustering algorithm.

Query-Ad Graph.

Given the set of related queries Q for a search query, we construct a query-ad graph as follows. First, we obtain all ads from the sponsored search data that match a query from Q: let the set of these ads be A. Then we construct a query-ad graph $G = (V, E)$, where $V = Q \cup A$ denotes the set of nodes in G, and E denotes the set of edges in G. In G, the edge between any two queries exists iff they co-occur in the same session; and the edge between any query-ad pair exists iff there is an impression record for that pair in the sponsored search data.

Transition Probability Matrix.

Once we have constructed the query-ad graph G, we prepare a transition matrix $\mathbf{P}$, which represents the transition probabilities among the nodes in G. $\mathbf{P}$ contains three types of transition probabilities: query-to-query, query-to-ad, and ad-to-ad. Figure 3 shows the transition model. As this figure shows, the model contains a parameter called ϵ, which determines the probability of transition from a query node to ad nodes.

query-to-query transition: This probability is determined based on the probability that a pair of queries co-occurs in the same session. The transition probability from query q_i to query q_j is defined as:

$$p(q_j|q_i) = (1-\epsilon)\,\frac{w_{\text{cooc}}(q_i, q_j)}{\sum_{q' \in R(q_i)} w_{\text{cooc}}(q_i, q')}\,,$$

where $w_{\text{cooc}}(q_i, q_j)$ denotes the number of sessions that q_i and q_j co-occurred, and $R(q_i)$ is the set of all queries that co-occurred with q_i within the same session. In practice, there are transitions from q_i to queries that are outside the set of related queries Q. We therefore introduce a special node f in G to collectively represent such queries, and we define the transition probability from q_i to f as follows:

$$p(f|q_i) = (1-\epsilon)\,\frac{\sum_{q' \in (R(q_i)-Q)} w_{\text{cooc}}(q_i, q')}{\sum_{q' \in R(q_i)} w_{\text{cooc}}(q_i, q')}\,.$$

That is, all transitions from each query q_i to queries outside Q are aggregated to node f. The node f has only the self-transition $p(f|f)$ with probability 1.

query-to-ad transition: This probability is defined as:

$$p(a_j|q_i) = \epsilon\,\frac{w_{\text{imp}}(q_i, a_j)}{\sum_{a' \in A} w_{\text{imp}}(q_i, a')}\,.$$

ad-to-ad transition: We define $p(a_j|a_i)$ to be 0 if $i \neq j$ and 1 if $i = j$.

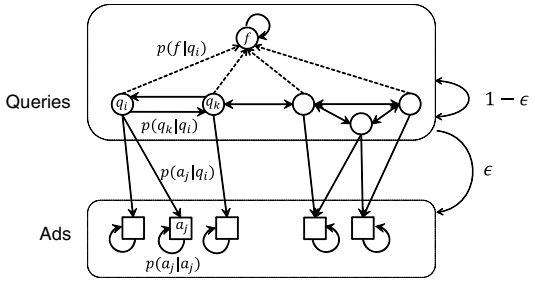


Figure 3: Example transitions on a query-ad graph. A random walk is applied on it for clustering queries.

Table 3: Data statistics.

(a) sponsored search data	
# of unique queries	25,394,581
# of unique ads	25,796,013
# of unique query-ad pairs	142,915,236
(b) clickthrough data	
# of unique queries	86,988,237
# of unique URLs	91,309,344
# of unique query-URL pairs	182,624,954
(c) session data	
# of queries	460,396,996
# of unique queries	142,253,653
# of sessions	214,396,433

Random Walk.

After preparing the transition probability matrix $\mathbf{P}$, we perform a random walk on the query-ad graph G. Let $\mathbf{P}^l$ be the transition probability matrix after an l-step random walk. The row in $\mathbf{P}^l$ for q_i can be interpreted as visit probabilities over the nodes in G after an l-step random walk that started at q_i.

As we can see from Figure 3, there is no ad-to-query transition, and there are only self-transitions among ads. Hence, for each q_i, the visit probabilities over Q approach zero while those over A converge as we iterate the random walk process. As Sadikov *et al.* notes, 3-5 iterations are enough in practice [28]. The transition probability matrix obtained after the convergence is referred to as $\mathbf{P}'$.

Query Clustering.

Finally, we cluster queries in Q using the transition probabilities from $\mathbf{P}'$. For each query $q_i \in Q$, its query vector is represented by a transition probability vector:

$$\mathbf{q}_i = [p'(a_1|q_i), \ldots, p'(a_j|q_i), \ldots, p'(a_{|A|}|q_i)]\,,$$

where $p'(a_j|q_i)$ denotes the transition probability from q_i to a_j in $\mathbf{P}'$, which can be interpreted as the probability that ad a_j would be displayed to the searcher who starts with query q_i. When two query vectors $\mathbf{q}_i$ and $\mathbf{q}_j$ are similar, this implies that similar ads are likely to be displayed in response to two different queries.

Following Sadikov *et al.* [28], we use a complete-linkage clustering method with cosine similarity of the above query vectors to obtain a set of k query clusters $\mathcal{C} = \{C_1, \ldots, C_k\}$.

4. EXPERIMENTAL SETUP

4.1 Logs

In our experiments, we used three types of logs, namely, sponsored search data, clickthrough data and session data. As was shown in Table 2, a sponsored search data record is a query-ad-impression triplet. Similarly, a clickthrough data record is a triplet composed of a query, clicked document and its click count. The latter is used for our implementation of the document-click-based

Table 4: Inter-assessor agreement on the number of annotated subgoals among the assessors (A1, A2, A3).

		A2			
		#0	#1	#2	Total
A1	#0	574	10	100	684
	#1	5	4	4	13
	#2	105	3	87	195
	Total	684	17	191	892

		A3			
		#0	#1	#2	Total
A1	#0	613	6	65	684
	#1	10	0	3	13
	#2	83	1	111	195
	Total	706	7	179	892

		A3			
		#0	#1	#2	Total
A2	#0	585	6	93	684
	#1	10	0	7	17
	#2	111	1	79	191
	Total	706	7	179	892

method of Sadikov *et al.* [28], which we use as a baseline in our experiments. As for session data, each record is a triplet composed of a session ID, a query and a timestamp, and a session is defined based on a 30-minute inactivity. All of these logs were sampled from a popular search engine and span exactly the same period from November 2011.

The raw sponsored search data contained some query-ad pairs for which queries and bid phrases were not *exact matches* (e.g. "health" vs. "health clubs"), thus we filtered out such pairs. This is because the query-ad pairs obtained through non-exact matches depend on the particular ads retrieval algorithm of the search engine, and we wanted to obtain results that are search engine independent. The statistics of the data thus obtained are shown in Table 3.

4.2 Proposed and Baseline Methods

To examine the effectiveness of our method that relies on sponsored search data, we implemented the following four methods.

(a) AdImp: This method clusters queries to mine subgoals as we described in Section 3. Given a search query, it obtains related queries from the sponsored search data, and then clusters them by combining ad impressions from the sponsored search data and query co-occurrences from the session data.

(b) AdImp (no cooc): This is the same as AdImp, except that query-to-query transitions derived from the session data are not used for clustering. That is, only query-to-ad transitions derived from the sponsored search data are utilized.

(c) DocClick: This is our implementation of the query clustering algorithm proposed by Sadikov *et al.* [28], which we treat as a baseline. Thus, given a search query, it obtains related queries from the session data, and then clusters them by combining document clicks from the clickthrough data and query co-occurrences from the session data. While Sadikov *et al.* originally obtained n most frequent queries that *follow* q_0 within the same session, we obtained n most frequent queries that *follow or is followed by* q_0 within the same session, as the order of issuing queries is not important for our purpose.

(d) DocClick (no cooc): This is the same as DocClick, except that query-to-query transitions derived from the session data are not used for clustering. That is, only query-to-document transitions derived from the clickthrough data are utilized.

Following Sadikov *et al.*, we set the number of related queries to $n = 80$ and the transition probability from query to ad or document to $\epsilon = 0.6$. Also, to construct the query-ad (or query-document) graph G for a given search query, we used only the top 15 most frequently displayed ads (or clicked documents) for each related query.

4.3 Test Collection Construction

In order to evaluate the effectiveness of our subgoal mining methods based on query clustering, we created our own test collections as described below.

4.3.1 Selecting Search Queries

Our first step was to select search queries that are to be used as input to our subgoal mining method. Thus, the input queries needed to be reasonably complex and to represent a search goal that may be associated with multiple subgoals. To select such input queries, we chose five domains (Business, Health, Recreation, Society and Sports) from the taxonomy used in the search engine, and extracted queries with high impressions from each domain[1]. From the high impression queries, we selected queries that contains at least one verb, or a noun that is *derivationally related* to a verb according to WordNet[2]. This is because we want to handle queries such as "weight loss" as well as explicitly verb-oriented queries like "lose weight." We thus obtained 892 candidate queries.

Three assessors independently annotated the above candidates to select input queries that are appropriate for subgoal mining. For each query, each assessor first judged whether it represents a search goal (i.e. a particular action that needs to be accomplished) either explicitly or implicitly; if it was judged as a search goal query, she wrote down up to two example subgoals in a *verb plus noun phrase* format (e.g. "do physical exercise" for a candidate query "lose weight").

Through the annotation task, 59 queries (6.6%) were annotated with two subgoals by all three assessors, and 349 queries (39.1%) were annotated with two subgoals by at least one assessor. Table 4 shows the inter-assessor agreement statistics on the number of annotated subgoals, where "#<number>" means the number of subgoals annotated by each assessor. The Fleiss' kappa [12] for this data set is 0.343, which is a moderate agreement. This is not altogether surprising, because given the same search queries, some people can think of good subgoals (i.e. subtasks or solutions), while others cannot. We are tackling the problem of subgoal mining precisely because we want to help the user by presenting possible subgoals that she may not be aware of.

From the annotated queries, we first selected those that were annotated with at least three unique subgoals, regardless of which assessor contributed them. Then, to increase the number of queries, we added some annotated queries for which at least one assessor identified two subgoals. Finally, we removed some queries from the set in order to avoid including very similar search goals. Through this selection process, we obtained a total of 125 search queries for our test collections, 25 queries for each of the five domains. Table 5 shows some example search queries. Note that although we selected these queries from the sponsored search data, these are also part of the session data, as these data sets were obtained from the same period.

4.3.2 Constructing Ground Truth Subgoals

In our present study, we view the problem of subgoal mining as a query clustering task. Thus, given a set of queries that are related to the search query, the problem is to cluster them appropriately, so that each query cluster represents an appropriate subgoal of the original search query. In order to evaluate this task, we need to build some ground truth data. We thus hired the same three assessors to *manually* cluster related queries and construct ground truth subgoals.

[1]Donato *et al.* reported that information needs in domains such as travel, health and education tend to be complex [11].

[2]WordNet, http://wordnet.princeton.edu/

Table 6: Statistics of each test collection. The mean and standard deviation are shown in the format of "<Mean>(<SD>)."

	Collection A			Collection B-1			Collection B-2		
	Both	Ads Only	Session Only	Both	Ads Only	Session Only	Both	Ads Only	Session Only
#search queries	98	–	–	23	–	–	23	–	–
#subgoals per query	9.65(2.80)	8.69(2.58)	7.77(2.49)	9.04(3.93)	7.30(3.37)	5.83(2.48)	9.87(2.29)	8.39(2.39)	7.48(2.26)
#Not Relevant per query	40.11(17.49)	6.59(7.18)	33.80(13.78)	75.30(17.10)	23.04(13.01)	52.91(11.22)	50.91(16.93)	9.13(6.92)	41.91(12.26)
#Not Subgoal per query	13.15(9.86)	7.89(7.57)	6.55(5.46)	30.74(16.55)	22.00(12.18)	12.43(8.80)	31.30(10.71)	22.13(8.48)	12.26(5.98)
Related query overlap per query	0.054(0.041)	–	–	0.059(0.042)	–	–	0.059(0.042)	–	–
Subgoal overlap per query	0.71(0.15)	–	–	0.51(0.24)	–	–	0.62(0.17)	–	–

Table 5: Example search queries in test collections.

Domain	Search query		
Business	car insurance project management	dept relief resume writing	lawn care
Health	back pain relief teeth whitening	eye care quit smoking	lose weight
Recreation	disney cruise vegas shows	fly fishing whale watching	hiking
Society	iq test us immigration	learn spanish wedding	sat prep
Sports	bodybuilding skateboarding	kayaking workout routines	hockey equipment

For each search query, we prepared a set of related queries by pooling the related queries obtained by AdImp and DocClick. Recall that the related queries of AdImp come from the sponsored search data, while those of DocClick come from the session data. As we want to compare the two approaches fairly, we included the related queries from both sides in order to manually identify possible subgoals.

We developed a simple GUI tool to facilitate the manual query clustering process, so that assessors could form clusters by dragging and dropping queries on the screen. Because not all of the pooled related queries represent a subgoal of the given search query, we prepared two special clusters called *Not Relevant* and *Not Subgoal*: the former was for queries that were topically non-relevant to the search goals, and the latter was for queries that are topically relevant but do not represent a subgoal. For example, for search query "lose weight," a related query "lose weight fast" should be put into the *Not Subgoal* cluster, as the latter is a specialization of the original query and does not represent a subgoal.

Using the GUI tool, the assessors manually clustered the displayed queries, and were also asked to provide a *subgoal label* in a "verb plus noun phrase" format. They were told that the target number of clusters was around 10, but were allowed to form fewer or more clusters if necessary. The tool also had a feature for assisting the assessor if she was unfamiliar with the related queries being displayed: by a right click on a related query, a web search result was shown to the user in a separate browser window.

Manual clustering is a tedious process: each assessor typically spent 30-60 minutes to complete the ground truth construction for one search query. In order to save the assessment cost while trying to maintain a reliable experimental environment, one assessor was assigned to 100 search queries, and the other two assessors were assigned to independently handle the remaining 25 search queries. Thus, we obtained two separate search query sets: the first one is relatively large but its ground truth data is constructed by only one assessor; the second one is relatively small but it has two sets of ground truth data.

4.3.3 Test Collection Statistics

As we mentioned above, we formed two separate sets of queries, one containing 100 and the other containing 25, and assigned one assessor to the former and two assessors to the latter. As a result of manual clustering, we found that two search queries from each query set did not contain enough relevant related queries, and therefore removed them. Thus, we obtained three subgoal mining test collections in the end: "Collection A" containing 98 search queries, as well as "Collection B-1" and "Collection B-2" sharing the same 23 search queries but annotated independently by two assessors. Using these three different test collections enables us to focus on general trends of the experimental outcome.

Table 6 shows the statistics of each test collection. The "Both" columns show statistics on the pooled related queries; "Ads Only" columns show statistics on the queries obtained from the sponsored search data (by the AdImp method); and "Session Only" columns shows those on the queries obtained from the session data (by the DocClick method). The "Related query overlap per query" shows the average of the Jaccard coefficient between the two sets of queries obtained from the sponsored search data and the session data. The "Subgoal overlap per query" shows the Jaccard coefficient between two subgoal sets that contain the queries obtained from the sponsored search data and those that contain the queries from the session data. In total, we obtained 1,375 ground truth subgoals (i.e. clusters) with 20,880 clustered related queries.

The "#Not Relevant" row of Table 6 indicates that many queries from the session data (obtained by the DocClick method) were classified as Not Relevant, i.e., off-topic. This suggests that sponsored search data may be a better resource than session data for obtaining topically related queries. On the other hand, the "#Not Subgoals" row shows that both sponsored search data and session data yield some related queries that are topically relevant but do not represent a subtopic of the input search query: recall the aforementioned "lose weight fast" example. In addition, through the analysis of "#Not Subgoals" queries obtained from the sponsored search data, we found some queries may be interpreted as *supergoals* of the given search query. For example, for the search query "kickboxing," related queries "lose weight" and "self defense" were obtained, due to the advertisers' effort in promoting kick-boxing lessons. That is, it is the *search query* "kickboxing" that can be interpreted as a subgoal of "lose weight" or "self defense."

The "Related query overlap per query" row shows that the overlap between the queries obtained based on the ad impressions in the sponsored search data and those obtained based on query co-occurrences in the session data is small. In contrast, the "Subgoal overlap per query" row shows that the overlap of identified subgoals between these two data sources is dramatically higher. This suggests that we may be able to obtain *searchers'* intents (often explicitly represented in session data) by leveraging the *advertisers'* efforts embedded in sponsored search data, even though the two data sources contain seemingly different queries.

As Collections B-1 and B-2 share the same input query set, we can measure inter-assessor agreement for the manually constructed ground truth clusters. For this, we compute the well-known F_1-measure, which measures how any given pair of items to be clustered is correctly grouped. Let TP denote the set of query pairs

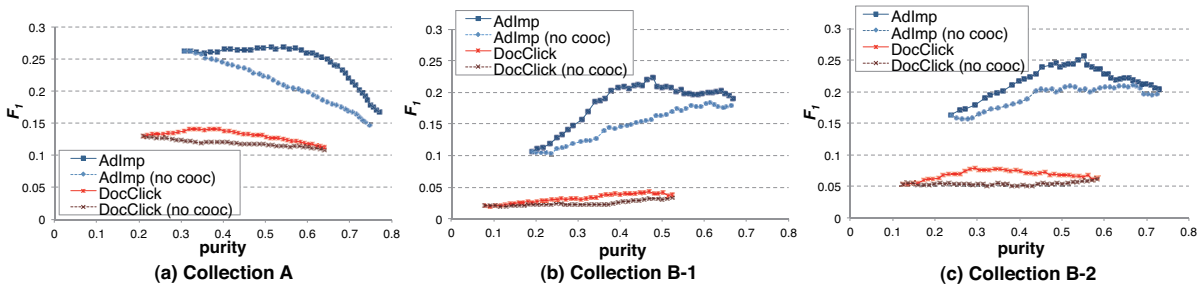


Figure 4: F_1 / purity graphs for the four methods, produced by varying the number of clusters k from 1 to 40.

from the same ground truth cluster that were correctly grouped together, FN denote the set for those that were incorrectly separated, and FP denote the set of query pairs from two different ground truth clusters but were incorrectly grouped together. Then $F_1 = \frac{2\mathrm{PR}}{\mathrm{P}+\mathrm{R}}$ where $\mathrm{P} = \frac{|\mathrm{TP}|}{|\mathrm{TP}|+|\mathrm{FP}|}$ and $\mathrm{R} = \frac{|\mathrm{TP}|}{|\mathrm{TP}|+|\mathrm{FN}|}$. When the subgoals from B-1 are treated as the ground truth, the F_1 of the subgoals from B-2 is 0.55. Conversely, when the subgoals from B-2 are treated as the ground truth, the F_1 of the subgoals from B-1 is 0.69. Thus the assessor agreement between B-1 and B-2 is reasonably high.

4.4 Evaluation Metrics

In addition to F_1, we also computed purity, Normalized Mutual Information (NMI) and Rand Index (RI) [25] to evaluate the quality of query clusters that represent subgoals. Purity, which we use as our primary metric along with F_1, measures the homogeneity of each cluster. Given n queries, a set of k clusters $\mathcal{C} = \{C_1, \ldots, C_k\}$ obtained by clustering the queries, and a set of l ground truth clusters that represent subgoals $\mathcal{S} = \{S_1, \ldots S_l\}$, purity is given by: $\frac{1}{n}\sum_{C_i \in \mathcal{C}} |C_i \cap S^{(C_i)}|$, where $S^{(C_i)} = \underset{S_j \in \mathcal{S}}{\operatorname{argmax}} |C_i \cap S_j|$, i.e. the "dominant" subgoal in C_i.

By definition, high purity can easily be achieved by choosing a high k (if every cluster contains exactly one query, then the purity is 1), while high F_1 can be achieved if k is close to the number of ground truth clusters. We thus plot F_1 against purity by varying k, to explore methods that achieve overall high accuracy *and* high within-cluster homogeneity.

In addition to the above well-known metrics, we also compute *subgoal recall* of a given cluster set $\mathcal{C}$ as: $\frac{|\cup_{C_i \in \mathcal{C}} S^{(C_i)}|}{|\mathcal{S}|}$.

In contrast to purity, this metric can penalize the case where multiple clusters correspond to the same dominant subgoal.

5. EXPERIMENTAL RESULTS

5.1 Results with All Related Queries

Figure 4 shows the F_1 / purity graphs for AdImp, AdImp (no cooc), DocClick and DocClick (no cooc), obtained by varying the number of clusters k from 1 to 40. Here, all of the related queries that were pooled (See Table 6) were included for evaluation. As for the Not Relevant and Not Subgoal queries in the ground truth data, each of them were treated as an independent cluster on its own when computing the metrics, as we are not interested in making the system form clusters out of these queries. From the figure, it can be observed that AdImp and AdImp (no cooc) achieve higher purity and F_1 compared to DocClick and DocClick (no cooc) for all three test collections. While our method uses an existing query clustering algorithm, it is clear from the results that our novel use of the ads data for the purpose of subgoal mining is highly effective. Moreover, it can be observed that utilizing the query co-occurrences from the session data for clustering is effective in both AdImp and DocClick.

AdImp

cheap wedding dress wedding dress wedding dresses wedding gowns	wedding cakes wedding decorations wedding favors wedding invitations	wedding bands wedding rings
		photographers wedding photographers
wedding planners wedding planning	dance studios	florists
	kate middleton	wedding gifts
wedding reception locations	wedding venues	

DocClick

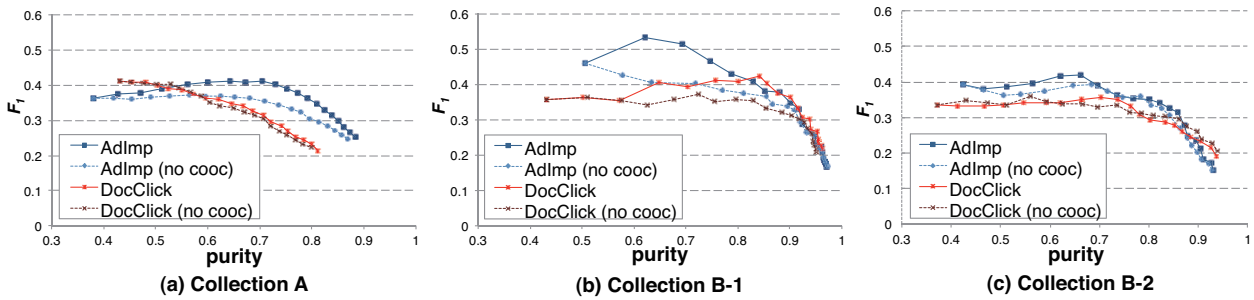
flowers gold love money video	engagement rings kim kardashian wedding tlc wedding rings wedding vows	davids bridal the knot wedding cakes wedding favors wedding invitations
hawaii photography	bed bath and beyond registry williams sonoma	father daughter dance

Figure 5: Query clusters produced by AdImp and DocClick, for the top 20 related queries of the search query "wedding."

As an example, Figure 5 contrasts some clusters obtained by AdImp and those obtained by DocClick for the search query "wedding." For each method, 80 related queries were clustered with the target number of clusters $k = 20$: only the top 20 high impression queries for AdImp and top 20 high co-occurrence queries for DocClick are shown in the figure. First, it can be observed that DocClick tends to produce more queries that (arguably) do not directly represent subgoals ("gold," "money," etc.) when compared to AdImp. Second, the clusters obtained by DocClick appear somewhat less homogeneous: for example, "kim kardashian wedding" ("kim kardashian" is the name of a celebrity) and "wedding ring" are in the same cluster. This happened because the query "kim kardashian wedding" shared the news article about her wedding with the query "wedding rings" in the clickthrough data. The query clusters for AdImp look somewhat more organized, thanks to the impressions of ads from various wedding services.

5.2 Results with Relevant Related Queries

While the results in Figure 4 appear to suggest that AdImp is much more effective than DocClick, it should be noted that there are at least two factors that may have contributed to the difference. The first is the quality of the related queries to be clustered: recall that while AdImp obtains related queries from the sponsored search data, DocClick (i.e. method by Sadikov *et al.*) obtains related queries from the session data, and the latter contains a lot of Not Relevant queries, as we have shown in Table 6. The second is the evidence we use for clustering: AdImp uses the ad impressions from the sponsored search data (with query co-occurrences from the session data), while DocClick uses the document clicks from the clickthrough data. Figure 4 does not show which of these factors are contributing by how much.



(a) Collection A (b) Collection B-1 (c) Collection B-2

Figure 6: F_1 / purity graphs for the four methods with relevant queries, produced by varying the number of clusters k from 1 to 20.

Table 7: Comparison of methods with different numbers of clusters k on Collection A, when only relevant related queries are clustered. Results that improved significantly (paired t-test) from DocClick are marked with "*" ($p < 0.05$) and "" ($p < 0.01$).**

		AdImp	AdImp (no cooc)	DocClick	DocClick (no cooc)
$k = 5$	purity	**0.560***	0.526	0.524	0.528
	NMI	**0.408****	0.341**	0.257	0.257
	RI	**0.598****	0.526**	0.452	0.433
	F_1	0.401	0.367	0.389	**0.402**
	subgoal recall	**0.381****	0.369*	0.336	0.339
$k = 10$	purity	**0.729****	0.681*	0.645	0.623
	NMI	**0.577****	0.518**	0.422	0.390
	RI	**0.750****	0.696**	0.601	0.550
	F_1	**0.402****	0.361	0.348	0.340
	subgoal recall	**0.569***	0.546	0.515	0.515

To separate the above two factors, Figure 6 shows the F_1 / purity graphs for the four methods *when only relevant related queries are clustered*, for k between 1 and 20. For each search query, let rel_{ad} and rel_{session} be the number of *relevant queries* (i.e. queries classified as neither Not Relevant nor Not Subgoal) obtained from the sponsored search data and the session data, respectively. We take $\min(rel_{\text{ad}}, rel_{\text{session}})$ relevant queries from both data sets, so that the contribution to the pool is equal in size for every search query. It can be observed that, even if we remove the effect of noise in the related queries to be clustered, AdImp generally achieves both high F_1 and high purity values.

Table 7 shows several metrics in the same "relevant queries only" setting for Collection A, when the number of required clusters k is 5 and 10. Significance differences with DocClick according to the paired t-test are indicated by asterisks, and the highest metric value among the four methods are indicated in bold. It can be observed, for example, that AdImp is significantly more effective than DocClick in terms of all metrics (purity, NMI, RI, F_1 and subgoal recall) when we require 10 clusters. Although not shown in this paper due to lack of space, the results for the other two test collections are generally similar.

These results suggest that the clustering step of AdImp, which relies on ad impressions, has advantages over that of DocClick, which relies on document clicks.

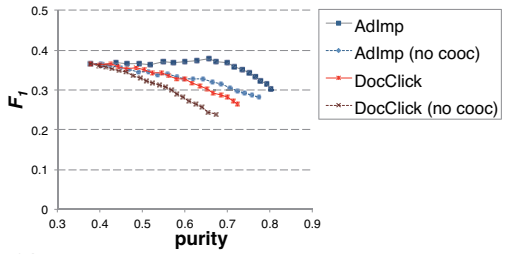
5.3 Results with Queries from the Same Source

The previous experiments showed that AdImp significantly outperforms DocClick, even if we remove the effect of noise in the related queries obtained from DocClick. Recall that while DocClick obtains related queries from the session data and then clusters them primarily based on document clicks from the clickthrough data, AdImp obtains related queries from the sponsored search data and then clusters them primarily based on ad impressions *from the same sponsored search data*. Thus, it is possible that the reason why AdImp works so well is that *queries from sponsored search data can be clustered accurately by leveraging the ad impressions from the same data*. In order to verify this hypothesis, we conducted two additional experiments: the first clustered only relevant queries that were obtained from the sponsored search data; the second clustered only relevant queries that were obtained from the session data.

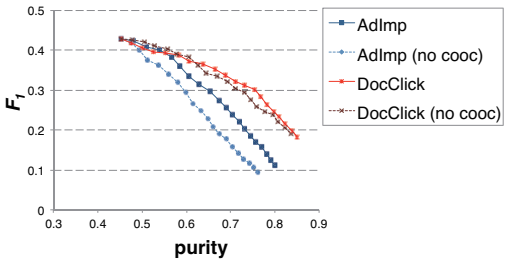
Figure 7 shows the F_1 / purity graphs for these two additional experiments on Collection A, for k between 1 and 20. First, Figure 7(a) shows that, for clustering related queries obtained from the sponsored search data, AdImp is clearly more effective than DocClick. Moreover, as the difference between AdImp and AdImp (no cooc) and that between DocClick and DocClick (no cooc) show, using the query co-occurrences from the session data at the clustering stage helps. On the other hand, Figure 7(b) shows that, AdImp is *less* effective than DocClick for clustering related queries obtained from the session data. This does not contradict with the above hypothesis (queries from sponsored search data can be clustered accurately by leveraging the ad impressions from the same data). The main reason why AdImp is less effective for queries from the session data is probably due to low coverage of sponsored search data: some queries have few or no associated ads. However, from the Figure 7 (b), while DocClick does not seem to benefit much from the use of query co-occurrences (compare with DocClick (no cooc)), the same statistics boost the performance of AdImp. This implies that session data can compensate for the low coverage of sponsored search data.

5.4 Ad Impressions vs. Ad Clicks

As we explained in Section 3.3, we calculated the query-to-ad transition probability based on *ad impressions*. One alternative way of calculating it is the use of *ad clicks*, which means the number of times ads were clicked by the search engine users. To compare the effect of ad impressions with ad clicks, we implemented two additional methods, namely, "AdClick" and "AdClick (no cooc)," both of which cluster the related queries as AdClick and AdClick (no cooc) do, except that both methods use the top 15 most *clicked ads* for constructing a query-ad graph, and calculate query-to-ad transition probability based on the number of times the ads were clicked. Figure 8 shows the results of these four methods, by varying the number of clusters k from 1 to 20 in the same "relevant query" setting in Section 5.2 for Collection A. From the figure, we can see that AdImp consistently outperforms AdClick, and AdImp (no cooc) also outperforms AdClick (no cooc) in terms of purity and F_1. This result is probably due to the sparseness of ad clicks. It is well known that CTR of ads are much less than those of documents (i.e. organic search results). Thus, it is hard to obtain reliable number of clicks for clustering related queries from the sponsored search data. Note, however, that even AdClick generally achieves both higher F_1 and purity than DocClick shown in Figure 6(a).



(a) Relevant queries from the sponsored search data



(b) Relevant queries from the session data

Figure 7: Comparison of ad impressions and document clicks under the same set of relevant queries on Collection A, for k between 1 and 20.

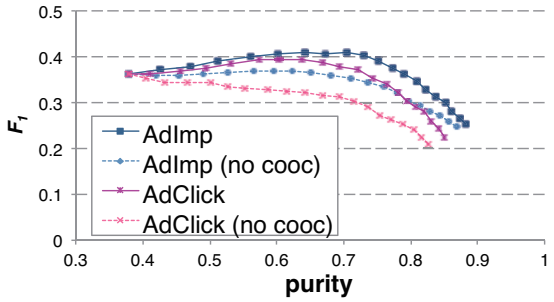


Figure 8: Comparison with ad impressions and clicks. The results were produced by AdImp on Collection A, for k between 1 and 20, when only relevant related queries are clustered.

5.5 Discussion

Our experiments showed that while AdImp generally outperforms DocClick, it is less effective for clustering related queries from the session data (See Section 5.3). As we discussed with Table 6, while the overlap of subgoals between queries from the sponsored search data and those from the session data is high, there are also subgoals that were obtained from only one of the data sources. On average across our three collections, 1.16 and 2.18 subgoals per search query were obtained from pure session data and from pure sponsored search data, respectively. In this subsection, we discuss some actual subgoals for these different cases.

Table 8 provides some examples from our three test collections. Table 8(a) shows subgoals that were formed only from the session data queries. As queries such as "causes of back pain" and "moving to do list" did not trigger any ads, the subgoals **learn about cause** and **make a TODO list** could not be obtained from the sponsored search data. Even if we increase the amount of sponsored search data, it is unlikely that our method will identify such subgoals as they are not strongly related to advertisement.

Table 8(b) shows a few example search queries for which both sponsored search data and session data were successful. Thus, both ad impressions and document clicks can successfully group queries like "moving truck" and "truck rental" together.

Table 8: Example of subgoals obtained: Each subgoal is a cluster of related queries with a manually assigned subgoal label.

(a) Subgoals that only contain queries from the session data.

Collection	B-2
search query	**back pain relief**
subgoal label	**learn about cause**
related queries	back pain causes, causes of back pain
Collection	B-1
search query	**moving**
subgoal label	**make a TODO list**
related queries	moving to do list, moving list of things to do, moving checklist, moving advice, ...

(b) Subgoals that contain queries obtained from both data sources.

Collection	B-1
search query	**learn spanish**
subgoal label	**build Spanish vocabulary**
related queries	learn spanish words, spanish dictionary, spanish verbs, spanish vocabrary, ...
Collection	B-2
search query	**moving**
subgoal label	**rent a truck**
related queries	moving truck, ryder, truck rental, ...

(c) Subgoals that only contain queries from the sponsored search data.

Collection	A
search query	**relieve stress**
subgoal label	**visit a zen garden**
related queries	zen garden
Collection	A
search query	**quit smoking**
subgoal label	**have acupuncture treatment**
related queries	acupuncture, acupuncture quit smoking, acupuncture stop smoking

Finally, Table 8(c) shows a few subgoals that were obtained only from the sponsored search data. Note that a search engine user who is looking for ways to "relieve stress" or to "quit smoking" may not even be aware that solutions such as **visit a zen garden** or **have acupuncture treatment** exist. Thus, it is the "wisdom of advertisers" that helps our method to *propose* such solutions to the searcher. It would be very difficult to find these "unexpected" solutions in the session data: if the searchers are unaware of these solutions, they are highly unlikely to issue queries about them.

6. CONCLUSIONS

In this study, we defined the problem of mining subgoals of a given search goal from data by means of query clustering, and proposed to utilize sponsored search data for this purpose. Our method (AdImp) first obtains related queries from the sponsored search data, and then clusters them based on ad impressions from the same data as well as query co-occurrences from session data. This was compared with a similar state-of-the-art method [28] (DocClick) that first obtains related queries from the session data, and then clusters them based on document clicks from clickthrough data as well as query co-occurrences from session data. Our experimental results using three in-house test collections showed that (1) related queries obtained from sponsored search data are more relevant than those obtained from session data; and (2) AdImp significantly outperforms DocClick in terms of purity, NMI, Rand Index, F_1 and subgoal recall.

There are several limitations to the present study. First, this study did not address the problem of generating a label for each query cluster (e.g. [26]). While our current representation of a subgoal is in the form of a cluster of several queries, it would be useful for the searcher if we could also provide an explicit label for each sub-

goal. We plan to tackle this problem by leveraging the ad contents: for example, the phrase "get some exercise" shown in Figure 1 is probably a good candidate as a subgoal label given the query "lose weight." Next, although we mentioned in Section 1 that around 68% of our head queries had some ads displayed with the organic search results, the applicability of our method is probably much lower than this number suggests. As we described in Section 4.3.1, only 39.1% of our candidate verb-oriented queries were annotated with multiple subgoals by at least one assessor. On a similar note, Jones and Klinkner [21], who defined the hierarchy of search missions and goals (See Section 2.4), also reported that only 20% of user queries from their Yahoo! query log were associated with hierarchically organized needs. Moreover, our method does not work if the query is not included in the sponsored search data, which is generally smaller compared to clickthrough and session data (See Table 3). However, although the fraction of queries that our method can handle may not be large, we argue that these queries represent a very important query segment, for which current search engines require breakthroughs.

As future work, we would like to refine our subgoal mining algorithm. As this study focused on verifying the usefulness of sponsored search data as a resource, we applied an existing query clustering algorithm. Thus, we did not have any explicit mechanisms, for example, for classifying subgoals of a given search query, *supergoals* of the query and queries that share the same goal with the query. Moreover, as we mentioned in Section 1, while some types of subgoal wholly accomplish the original search goal (e.g. "do physical exercise" may be a good solution to "lose weight"), others only partially accomplish the original search goal (e.g. "book flights" may be one step towards accomplishing the "travel London" search goal, but other subtasks such as "book a hotel" are also required). Furthermore, there may be temporal dependencies among these subtasks/subgoals: some subgoals need to be satisfied before others. Mining these different *types* of subgoal would be useful for improving search effectiveness and experience.

7. REFERENCES

[1] L. M. Aiello, D. Donato, U. Ozertem, and F. Menczer. Behavior-driven clustering of queries into topics. In *Proc. of CIKM*, pages 1373–1382, 2011.

[2] R. Baeza-Yates, C. Hurtado, and M. Mendoza. Query recommendation using query logs in search engines. In *Current Trends in Database Technology-EDBT 2004 Workshops*, pages 588–596, 2004.

[3] D. Beeferman and A. Berger. Agglomerative clustering of a search engine query log. In *Proc. of KDD*, pages 407–416, 2000.

[4] P. Boldi, F. Bonchi, C. Castillo, D. Donato, A. Gionis, and S. Vigna. The query-flow graph: model and applications. In *Proc. of CIKM*, pages 609–618, 2008.

[5] A. Broder. A taxonomy of web search. *ACM SIGIR Forum*, 36(2):3–10, 2002.

[6] A. Broder, P. Ciccolo, M. Fontoura, E. Gabrilovich, V. Josifovski, and L. Riedel. Search advertising using web relevance feedback. In *Proc. of CIKM*, pages 1013–1022, 2008.

[7] G. Buscher, S. T. Dumais, and E. Cutrell. The good, the bad, and the random: an eye-tracking study of ad quality in web search. In *Proc. of SIGIR*, pages 42–49, 2010.

[8] H. Cao, D. Jiang, J. Pei, Q. He, Z. Liao, E. Chen, and H. Li. Context-aware query suggestion by mining click-through and session data. In *Proc. of KDD*, pages 875–883, 2008.

[9] H. K. Dai, L. Zhao, Z. Nie, J.-R. Wen, L. Wang, and Y. Li. Detecting online commercial intention (OCI). In *Proc. of WWW*, pages 829–837, 2006.

[10] C. Danescu-Niculescu-Mizil, A. Z. Broder, E. Gabrilovich, V. Josifovski, and B. Pang. Competing for users' attention: on the interplay between organic and sponsored search results. In *Proc. of WWW*, pages 291–300, 2010.

[11] D. Donato, F. Bonchi, T. Chi, and Y. Maarek. Do you want to take notes?: identifying research missions in Yahoo! search pad. In *Proc. of WWW*, pages 321–330, 2010.

[12] J. Fleiss and J. Cohen. The equivalence of weighted kappa and the intraclass correlation coefficient as measures of reliability. *Educational and Psychological Measurement*, 1973.

[13] B. M. Fonseca, P. Golgher, B. Pôssas, B. Ribeiro-Neto, and N. Ziviani. Concept-based interactive query expansion. In *Proc. of CIKM*, pages 696–703, 2005.

[14] T. Graepel, J. Candela, T. Borchert, and R. Herbrich. Web-scale bayesian click-through rate prediction for sponsored search advertising in Microsoft's Bing search engine. In *Proc. of ICML*, pages 13–20, 2010.

[15] Q. Guo and E. Agichtein. Ready to buy or just browsing?: detecting web searcher goals from interaction data. In *Proc. of SIGIR*, pages 130–137, 2010.

[16] D. Hillard, S. Schroedl, E. Manavoglu, H. Raghavan, and C. Leggetter. Improving ad relevance in sponsored search. In *Proc. of WSDM*, pages 361–370, 2010.

[17] B. J. Jansen. The comparative effectiveness of sponsored and nonsponsored links for web e-commerce queries. *ACM Transactions on the Web*, 1(3), 2007.

[18] B. J. Jansen, D. Booth, and A. Spink. Determining the informational, navigational, and transactional intent of web queries. *Information Processing & Management*, 44(3):1251–1266, 2008.

[19] B. J. Jansen and M. Resnick. Examining searcher perceptions of and interactions with sponsored results. In *Workshop on Sponsored Search Auctions*, 2005.

[20] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems*, 20(4):422–446, 2002.

[21] R. Jones and K. L. Klinkner. Beyond the session timeout: automatic hierarchical segmentation of search topics in query logs. In *Proc. of CIKM*, pages 699–708, 2008.

[22] I. Kang and G. Kim. Query type classification for web document retrieval. In *Proc. of SIGIR*, pages 64–71, 2006.

[23] E. Kanoulas, B. Carterette, P. D. Clough, and M. Sanderson. Evaluating multi-query sessions. In *Proc. of SIGIR*, pages 1053–1062, 2011.

[24] Y. Liu, X. Ni, J.-T. Sun, and Z. Chen. Unsupervised transactional query classification based on webpage form understanding. In *Proc. of CIKM*, pages 57–66, 2011.

[25] C. D. Manning, P. Raghavan, and H. Schütze. *Introduction to Information Retrieval*. Cambridge University Press, 2008.

[26] M. P. Kato, T. Sakai, and K. Tanaka. Structured query suggestion for specialization and parallel movement: effect on search behaviors. In *Proc. of WWW*, pages 389–398, 2012.

[27] F. Radlinski, A. Broder, P. Ciccolo, E. Gabrilovich, V. Josifovski, and L. Riedel. Optimizing relevance and revenue in ad search: a query substitution approach. In *Proc. of SIGIR*, pages 403–410, 2008.

[28] E. Sadikov, J. Madhavan, L. Wang, and A. Halevy. Clustering query refinements by user intent. In *Proc. of WWW*, pages 841–850, 2010.

[29] X. Wang, D. Chakrabarti, and K. Punera. Mining broad latent query aspects from search sessions. In *Proc. of KDD*, pages 867–876, 2009.

[30] J.-R. Wen, J.-Y. Nie, and H.-J. Zhang. Clustering user queries of a search engine. In *Proc. of WWW*, pages 162–168, 2001.

[31] R. White and R. Roth. Exploratory search: Beyond the query-response paradigm. *Synthesis Lectures on Information Concepts, Retrieval, and Services*, 1(1):1–98, 2009.

[32] W. Xu, E. Manavoglu, and E. Cantu-Paz. Temporal click model for sponsored search. In *Proc. of SIGIR*, pages 106–113, 2010.