How Different Features Contribute to the Session Search?

Jingfei Li[1], Dawei Song[1,2], Peng Zhang[1](✉), and Yuexian Hou[1]

[1] Tianjin Key Laboratory of Cognitive Computing and Application,
Tianjin University, Tianjin, P.R. China
jingfl@foxmail.com, {dwsong,pzhang,yxhou}@tju.edu.cn
[2] The Computing Department, The Open University, Milton Keynes, UK

Abstract. Session search aims to improve ranking effectiveness by incorporating user interaction information, including short-term interactions within one session and global interactions from other sessions (or other users). While various session search models have been developed and a large number of interaction features have been used, there is a lack of a systematic investigation on how different features would influence the session search. In this paper, we propose to classify typical interaction features into four categories (current query, current session, query change, and collective intelligence). Their impact on the session search performance is investigated through a systematic empirical study, under the widely used Learning-to-Rank framework. One of our key findings, different from what have been reported in the literature, is: features based on current query and collective intelligence have a more positive influence than features based on query change and current session. This would provide insights for development of future session search techniques.

Keywords: Session features · Query change · Collective intelligence

1 Introduction

Session search aims to rank documents/web pages based on not only a current query, but also single or collective user interactions such as query reformulations and document clicks, in the current search session or longer-term search history [4][5][11][13][14]. Various session search models have been proposed [5][13][14] based on different interaction features extracted from specific resources.

However, it is still an open question how effectively different features contribute to improving the session search performance. This paper aims at a systematic investigation on this problem. We propose to classify typical interaction features into 4 categories (see Figure 1): (i) current query features; (ii) query change features; (iii) whole session features; and (iv) collective intelligence features. Each category is based on a specific assumption, namely *query relevance*, *search intent change*, *search intent relatedness*, and *collective intelligence helpfulness*, respectively. These assumptions (detailed in Section 3), individually,

J. Li et al. (Eds.): NLPCC 2015, LNAI 9362, pp. 242–253, 2015.
DOI: 10.1007/978-3-319-25207-0_21

are largely implied in different personalized search (including session search) approaches, e.g., [1][4][5][11][13]. We investigate these four assumptions (and the corresponding features) in one unified framework, i.e., the widely used Learning to Rank (Learning2Rank) framework. The most related work to ours is Bennett et al. [1] that investigated different user profiles built from users' long-term and short-term search behaviors. Our work is intrinsically different, in that we investigate session search based on different assumptions which reflect users' current Information Need (IN), evolving IN and collective intelligence, while Bennett et al. [1] studied the personalization based on features from different temporal views, i.e. historic view, session view and aggregate view. Moreover, our work is the first comparative study of the query change features against the other three types of features in the Learning2Rank framework.

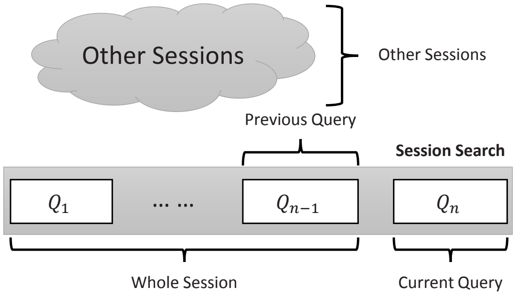


Fig. 1. An illustration of how we select features from current query, previous query, the whole session and other sessions with similar search goals to this session.

Specifically, after extracting different categories of features, we integrate them into the Learning2Rank model LambdaMART [2], then re-rank the original results returned by a baseline search engine. The re-ranking performance based on different features are compared. Experimental results on a real-world query log demonstrate that different categories of features have different impacts on re-ranking. One of our key findings is that the current query features and collective intelligence features are relatively more influential to the re-ranking performance. This provides new insights to the design of future session search models.

The remainder of this paper is organized as follows. Section 2 presents related work. Section 3 formalizes the background of this study. Section 4 introduces involved features in detail. Section 5 conducts extensive experiments to analyze how different features influence the session search. Conclusions and future work are discussed in Section 6.

2 Related Work

There are several lines of work related to ours including general personalized search and session search.

There have been many attempts to personalize the web search [4][11] and to investigate personalization-related IR problems (e.g., potential to personalization and personalization risks etc.) [1][9]. Dou et al. [4] proposed to re-rank the web search results with two personalization strategies (person-level and group-level), and found for the first time that personalization only works on a small subset of queries with larger click-entropy, this finding is also supported by many subsequent contributions [1][9][11]. Vu et al. [11] improved the search personalization with dynamic group of user profiles constructed in responds to the user's input query. Teevan [9] systematically studied the potential to personalization for queries with variations of features, and proposed that search personalization should be applied in different manners according to queries' characteristics. Bennett et al. investigates how different user profiles modeling user's long- or short- term search behaviors contribute to the re-ranking performance of web search results [1]. Similarly, Vu et al. also conducted an systematic investigation for personalization and built different user profiles on latent topic space from more fine grit temporal perspectives, i.e., long term, day term and session term [10]. Our work is similar to this two researches, but has intrinsic differences. We investigate the user's search behaviors based on different assumptions, while they studied the personalization based on different temporal views, i.e. historic view, session view and aggregate view. We integrate ranking features from different perspectives, e.g., query change and collective intelligence, rather than a single temporal angle, e.g., long-term and short-term.

Session search is a form of personalization utilizing users' short term interaction behaviors [5][7][13][14]. Guan et al. [5] and Zhang et al. [13] utilized query change information in session search which is based on the "search intent change" assumption introduced in Section 1. They listed some representative sessions from the TREC session tracks as examples to illustrate the phenomenon of user's query reformulation behaviors. They found that the current query of a session is composed of three parts, i.e. the common part, added part and removed part compared with previous query. Different weights are assigned to retrieved documents for three parts in ranking process. More specifically, the Markov Decision Process (MDP) was utilized to model the query change information in session search [5][13]. Luo et al. [7] and Zhang [14] also conducted session search considering the topic drifting and user's dynamics of information needs among queries within the same session. They model session search as a win-win game for the user and the search engine with the Partially Observable Markov Decision Process. Our work is inspired by them [5][7][13][14] which consider the dynamic information need of the user within a search session when retrieving documents. Differently, we focus on analyzing how effectively the query change features influence the overall retrieval performance within in a unified Learning-to-Rank framework, rather than developing a novel retrieval model.

3 Background

A session, formalized as $S =< q_1, ..., q_{n-1}, q_n >$, is a sequence of queries sorted by timestamp issued by one user. More general, a search session can be seen as a single search task or goal [5][14]. In session search, q_n is regarded as the current query whose original results are to be re-ordered. The re-ranking model may utilize the user's interactions with the search engine (recorded in previous queries $q_1, ..., q_{n-1}$), such as query issuing, query reformulation, results clicking, dwell time and paginating etc. [5][13]. To take advantage of the collective intelligence, some re-ranking models have incorporated other users' interaction information [4][10][11]. There exists so much information for session search and different information may interwine together, which makes session search a exceedingly challenging retrieval task.

LambdaMART [2] is a Learning2Rank model which has been proven to be effective for document ranking. For training and testing of the LambdaMART models, we design a semi-automated labeling algorithm for estimating the relevance of each document with regard to a query. The SAT-clicked documents (with dwell time more than 30 seconds [1][11]) are labeled as 2, the clicked documents with a dwell time less than 30 seconds are labeled as 1 and the other documents are labeled as 0. This dwell time based grading method is also used in [14].

4 A Classification of Interaction Features

In this paper, we classify the typical interaction features into four categories (i.e., current query features, query change features, whole session features, and collective intelligence features), based on four underlying assumptions. First, the "current query" features are underpinned by the *query relevance* assumption. It assumes that a user-issued query represents the user's Information Need (IN) directly. However, the representation is incomplete due to the limitation of user knowledge. Therefore, more information (features) needs to be imported to enrich the representation of user IN. Second, the *search intent change* assumption, which underpins the "query change" features, is based on the fact that user's IN evolves continuously. The query reformulation information between the current and previous queries provides some clues to capture these changes. Third, the *search intent relatedness* assumption, which is related to the "whole session" features, considers that queries in the same session have a similar search intent, and the previous queries and clicked documents in the same session can reveal the search intent of current query to some extent. Finally, *collective intelligence helpfulness* underpinning the "collective intelligence" features, assumes that the interactions of other people, especially the ones who have similar search interests or have submitted similar queries in the past, will provide useful clues for ranking documents for the current user.

In the rest of this section, we will describe each category of features in detail. An empirical comparative study is then reported in Section .

4.1 Current Query Features

This category of features only consider the similarity of a document d to the current query q. Three traditional scoring schemes which have been proven to be effective are included in this category. They are BM25 [8][1], Query Likelihood model [12][2] and $tf \cdot idf$ based ranking function [6]. Additionally, the ranks of documents in the original result list for the current query q is also considered. A score is estimated as $frank(q,d) = \frac{1}{log_2(1+rank_d)}$, where $rank_d$ is the rank of document d. The current query features are summarized as Table 1-(a). The computation equations for different features are formalized as follows.

One of the most prominent instantiation among the whole family of BM25 based ranking functions is formulated as follows:

$$bm25(d,q) = \sum_{w \in q} idf(w) \cdot \frac{c(w,d) \cdot (k_1+1)}{c(w,d) + k_1 \cdot (1 - b + b \cdot \frac{|d|}{avgdl})} \cdot \frac{(k_3+1) \cdot c(w,q)}{k_3 + c(w,q)} \tag{1}$$

where $c(w,d)$ and $c(w,q)$ are w's term frequencies in d and q respectively; $avgdl$ is the average document length of the text collection. $idf(w) = log\frac{N-df_w+0.5}{df_w+0.5}$, where N is the total document count in a collection, df_w is the number of documents containing word w; k_1, b and k_3 are three parameters which, in this paper, are empirically set as 1.2, 0.75 and 7 respectively.

The query likelihood model is a language model used in IR, and it can be interpreted as being the likelihood of a document being relevant to a query. The relevance score based on query likelihood is as follows:

$$QL(q|\theta_d) = \prod_{w \in q} p(w|d) \propto \sum_{w \in q} \log p(w|d),\ p(w|d) = \frac{c(w,d) + \mu p(w|C)}{|d| + \mu} \tag{2}$$

where θ_d is a unigram language model, i.e., $\theta_d = p(w|d)_{w \in V}$, V is the vocabulary, $p(w|d)$ is the probability of word w. In this paper we use Dirichlet prior smoothing method to estimate the probability of w, where $p(w|C)$ is a language model for the collection and the smoothing parameter $\mu = 2500$ here.

We follow the definition of $tf \cdot idf$ score of a document d given a query q as Liu et al.[6] does. $c(w,d)$ is the term frequency of w in document d, df_w is the document frequency of w, and $|C|$ is the total count of documents in collection.

$$tfidf(q,d) = \sum_{w \in q \cap d} (0.5 + \frac{0.5 \times c(w,d)}{max\{c(w,d) : w \in d\}}) log\frac{|C|}{df_w} \tag{3}$$

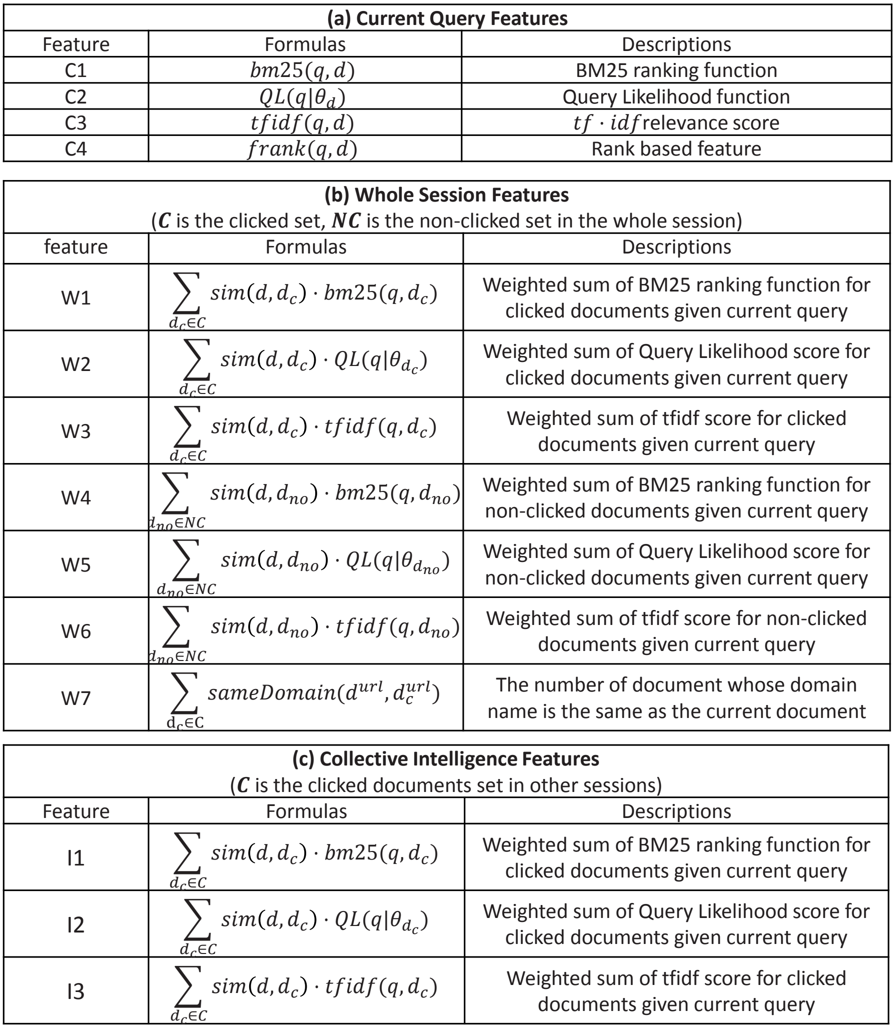
4.2 Query Change Features

Query change, also known as query reformulation, is an important type of user interaction with the search engine. Lacking satisfaction with the returned results for an initial query, the user may change the query to achieve her/his search target. How to reformulate the query reflects the change direction of the user's

[1] The well known parameters k_1, b and k_3 for Okapi BM25 are empirically set as 1.2, 0.75 and 7 respectively in this paper.

[2] The Dirichlet prior smoothing method is used to estimate the probability of a word, and the smoothing parameter $\mu = 2500$ here.

Table 1. Features and their description. (a) is the current query dependent features; (b) is the whole session features; (c) is the collective intelligence features. d_c and d_{no} denote the clicked documents and non-clicked documents respectively.



(a) Current Query Features

Feature	Formulas	Descriptions
C1	$bm25(q,d)$	BM25 ranking function
C2	$QL(q\|\theta_d)$	Query Likelihood function
C3	$tfidf(q,d)$	$tf \cdot idf$relevance score
C4	$frank(q,d)$	Rank based feature

(b) Whole Session Features
(***C*** is the clicked set, ***NC*** is the non-clicked set in the whole session)

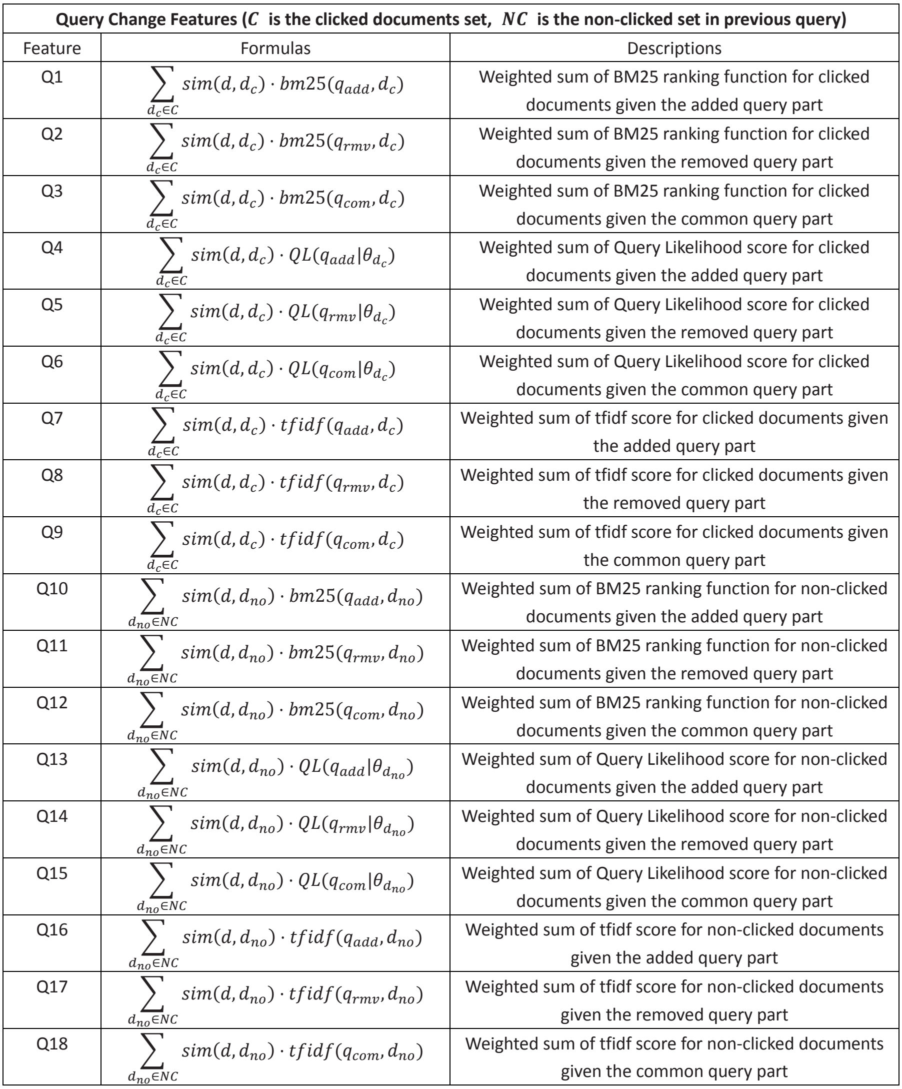
feature	Formulas	Descriptions
W1	$\sum_{d_c \in C} sim(d,d_c) \cdot bm25(q,d_c)$	Weighted sum of BM25 ranking function for clicked documents given current query
W2	$\sum_{d_c \in C} sim(d,d_c) \cdot QL(q\|\theta_{d_c})$	Weighted sum of Query Likelihood score for clicked documents given current query
W3	$\sum_{d_c \in C} sim(d,d_c) \cdot tfidf(q,d_c)$	Weighted sum of tfidf score for clicked documents given current query
W4	$\sum_{d_{no} \in NC} sim(d,d_{no}) \cdot bm25(q,d_{no})$	Weighted sum of BM25 ranking function for non-clicked documents given current query
W5	$\sum_{d_{no} \in NC} sim(d,d_{no}) \cdot QL(q\|\theta_{d_{no}})$	Weighted sum of Query Likelihood score for non-clicked documents given current query
W6	$\sum_{d_{no} \in NC} sim(d,d_{no}) \cdot tfidf(q,d_{no})$	Weighted sum of tfidf score for non-clicked documents given current query
W7	$\sum_{d_c \in C} sameDomain(d^{url},d_c^{url})$	The number of document whose domain name is the same as the current document

(c) Collective Intelligence Features
(***C*** is the clicked documents set in other sessions)

Feature	Formulas	Descriptions
I1	$\sum_{d_c \in C} sim(d,d_c) \cdot bm25(q,d_c)$	Weighted sum of BM25 ranking function for clicked documents given current query
I2	$\sum_{d_c \in C} sim(d,d_c) \cdot QL(q\|\theta_{d_c})$	Weighted sum of Query Likelihood score for clicked documents given current query
I3	$\sum_{d_c \in C} sim(d,d_c) \cdot tfidf(q,d_c)$	Weighted sum of tfidf score for clicked documents given current query

search intent. Existing researches [5][13][14] analyzed different strategies of query reformulation from two aspects, i.e., query formation and query semantic. In this paper, we extract query change features based on the change of query formation, including adding and removing query terms. To this end, we segment the current query into three parts, i.e., the common part (obtained by $q_{com} = wordset(q_n) \cap wordset(q_{n-1})$), added part ($q_{add} = wordset(q_n) - q_{com}$) and removed part ($q_{rmv} = wordset(q_{n-1}) - q_{com}$) compared with previous query. The query-document features are computed based on the weighed sum of

Table 2. Query change features and their description. d_c and d_{no} denote the clicked documents and non-clicked documents respectively.



Query Change Features (C is the clicked documents set, NC is the non-clicked set in previous query)		
Feature	Formulas	Descriptions
Q1	$\sum_{d_c \in C} sim(d, d_c) \cdot bm25(q_{add}, d_c)$	Weighted sum of BM25 ranking function for clicked documents given the added query part
Q2	$\sum_{d_c \in C} sim(d, d_c) \cdot bm25(q_{rmv}, d_c)$	Weighted sum of BM25 ranking function for clicked documents given the removed query part
Q3	$\sum_{d_c \in C} sim(d, d_c) \cdot bm25(q_{com}, d_c)$	Weighted sum of BM25 ranking function for clicked documents given the common query part
Q4	$\sum_{d_c \in C} sim(d, d_c) \cdot QL(q_{add} \vert \theta_{d_c})$	Weighted sum of Query Likelihood score for clicked documents given the added query part
Q5	$\sum_{d_c \in C} sim(d, d_c) \cdot QL(q_{rmv} \vert \theta_{d_c})$	Weighted sum of Query Likelihood score for clicked documents given the removed query part
Q6	$\sum_{d_c \in C} sim(d, d_c) \cdot QL(q_{com} \vert \theta_{d_c})$	Weighted sum of Query Likelihood score for clicked documents given the common query part
Q7	$\sum_{d_c \in C} sim(d, d_c) \cdot tfidf(q_{add}, d_c)$	Weighted sum of tfidf score for clicked documents given the added query part
Q8	$\sum_{d_c \in C} sim(d, d_c) \cdot tfidf(q_{rmv}, d_c)$	Weighted sum of tfidf score for clicked documents given the removed query part
Q9	$\sum_{d_c \in C} sim(d, d_c) \cdot tfidf(q_{com}, d_c)$	Weighted sum of tfidf score for clicked documents given the common query part
Q10	$\sum_{d_{no} \in NC} sim(d, d_{no}) \cdot bm25(q_{add}, d_{no})$	Weighted sum of BM25 ranking function for non-clicked documents given the added query part
Q11	$\sum_{d_{no} \in NC} sim(d, d_{no}) \cdot bm25(q_{rmv}, d_{no})$	Weighted sum of BM25 ranking function for non-clicked documents given the removed query part
Q12	$\sum_{d_{no} \in NC} sim(d, d_{no}) \cdot bm25(q_{com}, d_{no})$	Weighted sum of BM25 ranking function for non-clicked documents given the common query part
Q13	$\sum_{d_{no} \in NC} sim(d, d_{no}) \cdot QL(q_{add} \vert \theta_{d_{no}})$	Weighted sum of Query Likelihood score for non-clicked documents given the added query part
Q14	$\sum_{d_{no} \in NC} sim(d, d_{no}) \cdot QL(q_{rmv} \vert \theta_{d_{no}})$	Weighted sum of Query Likelihood score for non-clicked documents given the removed query part
Q15	$\sum_{d_{no} \in NC} sim(d, d_{no}) \cdot QL(q_{com} \vert \theta_{d_{no}})$	Weighted sum of Query Likelihood score for non-clicked documents given the common query part
Q16	$\sum_{d_{no} \in NC} sim(d, d_{no}) \cdot tfidf(q_{add}, d_{no})$	Weighted sum of tfidf score for non-clicked documents given the added query part
Q17	$\sum_{d_{no} \in NC} sim(d, d_{no}) \cdot tfidf(q_{rmv}, d_{no})$	Weighted sum of tfidf score for non-clicked documents given the removed query part
Q18	$\sum_{d_{no} \in NC} sim(d, d_{no}) \cdot tfidf(q_{com}, d_{no})$	Weighted sum of tfidf score for non-clicked documents given the common query part

basic ranking scores (e.g., BM25, Query Likelihood and tfidf scores) of clicked documents in previous query given the three query parts respectively. For example, the cumulative okapi BM25 score of a document given the common part is formalized as $cumulative\_bm25(q_{com}, d) = \sum_{d_c \in C} sim(d, d_c) \times bm25(q_{com}, d_c)$, where d_c is a clicked document in the clicked documents set C of previous query, $sim(d_1, d_2)$ is the Cosine similarity between two documents represented with $tf \cdot idf$ vectors. The query change features are summarized as Table 2.

4.3 Whole Session Features

The features of this category are similar to the short-term features in Bennett et al. [1]. Although the information needs for different queries in the same session vary, they are supposed to somehow relate to the current query. For instance, one may issue a query to ask about the basic information of one city, and then issue another related query about the representative historical figures or famous scenery spots. Utilizing the previous related queries for session search may help to disambiguate the current query. We obtain the whole session features by computing the cumulative basic scores based on all clicked documents in previous queries in the same session. The features are summarized in Table 1-(b).

4.4 Collective Intelligence Features

Massive attempts on search personalization have shown that the performance of IR models can be improved by enriching global information related to current user [4][11]. It is also a popular phenomenon that our searching problems have been solved by others. Our searching behaviors may also be inspired by some popular events. Therefore integrating the collective intelligence features in IR is a natural choice in our investigation. We extract features from the clicked documents in other sessions topically similar to the current session. To this end, we utilize the Latent Dirichlet Allocation (LDA) to learn topics from clicked documents in the query log. Let T, W and D be variables which represent a latent topic, a single word, and a document respectively. The session variable is denoted as S. The instances of T, W, D and S are denoted as t, w, d, and s. $P(W|T)$ corresponds to a distribution of words for each topic, which shows the relevant probability of a word to the topic. $P(T|D)$ corresponds to a distribution of these learned latent topics for each document, which shows the probability a topic is relevant to the document. Based on the trained topics, we define the probability of a session s being relevant to a learned latent topic t as a conditional probability [11]:

$$p(t|s) = \frac{1}{|C(s)|} \sum_{d \in C(s)} p(t|d) \tag{4}$$

where $C(s)$ is the set of clicked documents in session s, and $p(t|d)$ is the probability of topic t given the document d. In this way, a session s can be represented as a vector of topics, denoted VT_s. Formally, $VT_s =< p(t_1|s), p(t_2|s), ..., p(t_{|T|}|s) >$, where t_i is the i_{th} latent topic, $|T|$ is the number of latent topics. We select top K sessions to extract the collective intelligence features according to the Cosine similarity between the topic vectors of current session and other sessions. The features of this category are summarized in the Table 1-(c).

5 Empirical Comparison of Different Features

5.1 Experimental Setup

Our experiments have been conducted on a query log containing 489,384 queries[3]. Each query in the log comes with the issued timestamp, anonymous user-id, clicked URLs, dwell time on clicked URLs and a list of URLs returned by the search engine. We segmented the query log into 126,103 sessions according to some simple but widely accepted criteria, i.e., the time interval between two subsequent sessions of a user are more than 30 minutes and the queries within one session are sorted by their issued timestamps [1]. Figure 2 reports the distribution of query number and session number on dates, which shows that the distributions are relatively uniform over all active days. However, massive existing work has demonstrated that there is little potential to personalization for queries with click entropy[4] larger than some threshold [1][4][11]. Thus, in our experiments, we only selected a part of sessions, in which the current queries' click entropies are larger than 2.

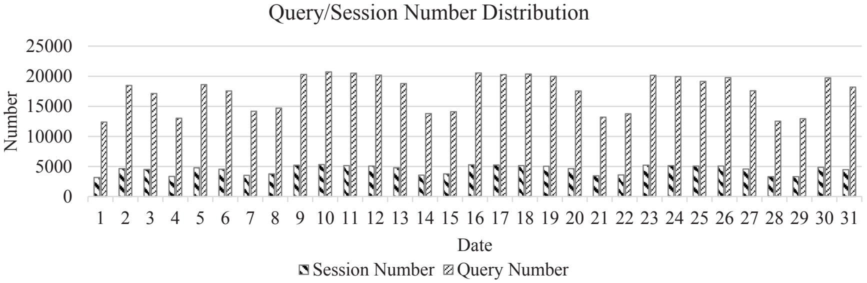


Fig. 2. Distribution of query (session) number on dates.

The RankLib[5] is utilized to run the LambdaMART algorithm, in which "-norm" is set as "zscore", all LambdaMART-specific parameters (e.g., "-tree" and "-leaf") are set as default values. The selected sessions in the first 3 weeks are

[3] The query log is collected from the Bing search engine in 4 weeks (from July 1^{th} 2012 to July 28^{th} 2012) for 1166 users. All queries are from the US market, non-English queries are filtered out.

[4] Click Entropy[4] is a direct indication of query click variation, less click entropy means more focus of URLs on a query. It is defined as follows: $ClickEntropy(q) = \sum_{u \in U(q)} -P(u|q) \log_2 P(u|q)$, where $U(q)$ is the set of web pages (URLs) that are clicked with respect to the distinct query q, and $P(u|q)$ is the percentage of the clicks on URL u among all the clicks for the query q.

[5] https://sourceforge.net/p/lemur/wiki/ranklib/

randomly partitioned into a training set (2663 sessions) and a validation set (203 sessions). All selected sessions in the last week are set as the test set (851 sessions). The target metrics are respectively set as "ERR@10" and "NDCG@10" corresponding to two evaluation metrics used in this study.

We set the original results ("ORI") given by the search engine as the baseline model. Given that our aim is to investigate how different categories of features contribute to the session search performance, we design different strategies (considering different feature groups) to train and test the LambdaMART ranking models, which are list as follows.

1. CUR, only current query features are considered by the learner;
2. CHA, only query change features are considered by the learner;
3. WHO, only the whole session features are considered by the learner;
4. COL, only the collective intelligence features are considered;
5. ALL, all features are considered by the learner;
6. AECUR, all features except for current query features are considered;
7. AECHA, all features except for query change features are considered;
8. AEWHO, all features except for whole session features are considered;
9. AECOL, all features except for collective intelligence features;

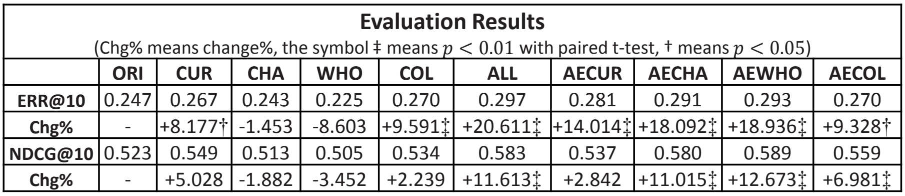
Note that, it is important to determine the number K of sessions selected to extract the collective intelligence features. We conducted a series of pilot experiments and eventually selected $K = 2$, which gained the best performance.

5.2 Results and Analysis

We adopt ERR@10 [3] and NDCG@10 [8] as the evaluation metrics (and as target in training). Table 3 reports the evaluation results of ranking models considering different categories of features.

As illustrated in Table 3, when only one category of features is used (by learner), CUR and COL have better re-ranking performance than CHA and WHO. This shows that the current query features and collective intelligence features have more positive influence than the query change features and the whole session features. It is in accordance with our expectation that the current query features are important since the current query represents user's IN directly. We also find that models considering collective intelligence features outperform

Table 3. Experimental results evaluated with ERR@10 and NDCG@10. Rows Chg% report the change percentage of evaluation metrics compared with the baseline.



Evaluation Results

(Chg% means change%, the symbol ‡ means $p < 0.01$ with paired t-test, † means $p < 0.05$)

	ORI	CUR	CHA	WHO	COL	ALL	AECUR	AECHA	AEWHO	AECOL
ERR@10	0.247	0.267	0.243	0.225	0.270	0.297	0.281	0.291	0.293	0.270
Chg%	-	+8.177†	-1.453	-8.603	+9.591‡	+20.611‡	+14.014‡	+18.092‡	+18.936‡	+9.328†
NDCG@10	0.523	0.549	0.513	0.505	0.534	0.583	0.537	0.580	0.589	0.559
Chg%	-	+5.028	-1.882	-3.452	+2.239	+11.613‡	+2.842	+11.015‡	+12.673‡	+6.981‡

those considering the query change features and whole session features. We consider this finding meaningful, as it deviates from the observations in the existing work and may benefit the design of future session search algorithms. A possible interpretation of this phenomenon is that the topically similar sessions provide very useful information for current search task. To our best knowledge, we are the first to discover this phenomenon that other similar sessions' features have more positive influence than current session features including the query change features and the whole session features.

Ranking models considering multiple categories of features outperform all models that only consider a single category features. This illustrates that the combination of different categories features can improve the effectiveness of session search. Moreover, various feature combinations have different influences on re-ranking performance. With regard to ERR@10, the best feature combination is ALL, and removing any category of features will hurt the effectiveness of session search to different degrees. For NDCG@10, AEWHO is the best performing combination which outperforms ALL. This reflects that the whole session features may have some negative influences on session search. Removing current query features or collective intelligence features have more impact than query change features and the whole session features, with respect to both ERR@10 and NDCG@10.

6 Conclusions and Future work

In this paper, we have classified different interaction features for session search into four categories. We then trained and tested a series of session models considering different categories of features. Experimental results show that the current query features and collective intelligence features have more positive influence on re-ranking performance than query change features and whole session features. Our findings will potentially bring benefits for the design of future information retrieval models which can take full advantages of the collective intelligence besides the features extracted from the current query.

Although, in this paper, query change features did not gain a good performance, we consider this category of interactions very important to detect user's evolving IN in exploratory search, thus worth further investigating in the future. To our best knowledge, we are the first to integrate the query change information into the Learning2Rank framework, and we are the first to explicitly formalize these four categories of features together. In the future, we believe that the integration of multidimensional features which can reflect the dynamics of users' information need within a search session will be promising research topic. Additionally, in order to have a better understanding on how different features work on session search, the analysis of retrieval performances on different sessions (e.g., with different queries) could be conducted in the future.

Acknowledgments. The work presented in this paper is sponsored in part by the Chinese National Program on Key Basic Research Project (973 Program, grant No.

2013CB329304, 2014CB744604), the Chinese 863 Program (grant No. 2015AA015403), the Natural Science Foundation of China (grant No. 61402324, 61272265), and the Research Fund for the Doctoral Program of Higher Education of China (grant no. 20130032120044).

References

1. Bennett, P.N., White, R.W., Chu, W., Dumais, S.T., Bailey, P., Borisyuk, F., Cui, X.: Modeling the impact of short-and long-term behavior on search personalization. In: SIGIR, pp. 185–194. ACM (2012)
2. Burges, C.J.: From ranknet to lambdarank to lambdamart: An overview. Learning **11**, 23–581 (2010)
3. Chapelle, O., Metlzer, D., Zhang, Y., Grinspan, P.: Expected reciprocal rank for graded relevance. In: CIKM, pp. 621–630. ACM (2009)
4. Dou, Z., Song, R., Wen, J.-R.: A large-scale evaluation and analysis of personalized search strategies. In: WWW, pp. 581–590. ACM (2007)
5. Guan, D., Zhang, S., Yang, H.: Utilizing query change for session search. In: SIGIR, pp. 453–462. ACM (2013)
6. Liu, T.-Y., Xu, J., Qin, T., Xiong, W., Li, H.: Letor: benchmark dataset for research on learning to rank for information retrieval. In: Proceedings of SIGIR 2007 Workshop on Learning to Rank for Information Retrieval, pp. 3–10 (2007)
7. Luo, J., Zhang, S., Yang, H.: Win-win search: dual-agent stochastic game in session search. In: SIGIR, pp. 587–596. ACM (2014)
8. Manning, C.D., Raghavan, P., Schütze, H.: Introduction to information retrieval, vol. 1. Cambridge University Press Cambridge (2008)
9. Teevan, J., Dumais, S.T., Horvitz, E.: Potential for personalization. TOCHI **17**(1), 4 (2010)
10. Vu, T., Willis, A., Tran, S.N., Song, D.: Temporal latent topic user profiles for search personalisation. In: Hanbury, A., Kazai, G., Rauber, A., Fuhr, N. (eds.) ECIR 2015. LNCS, vol. 9022, pp. 605–616. Springer, Heidelberg (2015)
11. Vu, T., Song, D., Willis, A., Tran, S.N., Li, J.: Improving search personalisation with dynamic group formation (2014)
12. Zhai, C.: Statistical language models for information retrieval. Synthesis Lectures on Human Language Technologies **1**(1), 1–141 (2008)
13. Zhang, S., Guan, D., Yang, H.: Query change as relevance feedback in session search. In: SIGIR, pp. 821–824. ACM (2013)
14. Zhang, S., Luo, J., Yang, H.: A pomdp model for content-free document re-ranking. In: SIGIR, pp. 1139–1142. ACM (2014)