Modelling Time-aware Search Tasks for Search Personalisation

Thanh Vu, Alistair Willis, and Dawei Song*
The Open University, Milton Keynes, UK
Tianjin University, Tianjin, P.R.C*
firstname.surname@open.ac.uk

ABSTRACT

Recent research has shown that mining and modelling search tasks helps improve the performance of search personalisation. Some approaches have been proposed to model a search task using topics discussed in relevant documents, where the topics are usually obtained from human-generated online ontology such as Open Directory Project. A limitation of these approaches is that many documents may not contain the topics covered in the ontology. Moreover, the previous studies largely ignored the dynamic nature of the search task; with the change of time, the search intent and user interests may also change.

This paper addresses these problems by modelling search tasks with time-awareness using latent topics, which are automatically extracted from the task's relevance documents by an unsupervised topic modelling method (i.e., Latent Dirichlet Allocation). In the experiments, we utilise the time-aware search task to re-rank result list returned by a commercial search engine and demonstrate a significant improvement in the ranking quality.

Categories and Subject Descriptors: H.3.3 [Information Systems Applications]: Information Search and Retrieval

Keywords: Time-aware Search Task; Search Personalisation; Latent Topics

1. INTRODUCTION

Recently, search personalisation has been an active research area and attracted increasing attention in literature [1, 2, 4, 5]. Previous research has shown that mining and modelling *search task*, which represents an atomic user information need [2], helps improve the performance of web search personalisation [2, 5]. In the context of the web search logs, a user may submit several queries within a search task and handle several tasks within a *search session* [2].

A common approach is to model the search task with the main topics discussed in the task's relevant documents [2, 5]. In previous work, the topics of a document have often obtained from a human-generated online ontology, such as the Open Directory Project (ODP). One limitation of these approaches is that many documents may not contain the topics covered in the ontology. Furthermore, it needs expensive manual effort to determine the correct categories for each document [4]. In addition, the dynamic nature of the search task is largely ignored in previous search task modelling studies (e.g., search task intent may be generalised or specialised over the searching time).

To handle these problems, in this paper, we propose a unified framework to model search tasks which evolve from time to time over a topic space (in Section 2.1). We utilise latent topics automatically derived from the task's relevant documents by an unsupervised topic modelling method (i.e., Latent Dirichlet Allocation (LDA)) instead of using a human-generated ontology [2, 5].

2. PERSONALISATION FRAMEWORK

2.1 Modelling Time-aware Search Tasks

Clustering Search Tasks in Sessions The primary source of data for this study is a query logs from a commercial search engine. A log sample consists of an anonymous user identifier, a query, top-10 returned URLs, and clicked results along with the user's dwell time. To identify a session, we use the common approach of demarcating session boundaries by 30 minutes of user inactivity [1].

To identify tasks from each search session, we apply the Query Task Clustering approach (QTC) [2], which has a desirable feature of extracting interleaved tasks within a session. In general, the method works in two steps: first, measure the similarities between query pairs to build an undirected graph of queries within each search session; second, cluster queries into tasks by dropping the weak edges where the similarities are smaller than a threshold (0.9 in our case). Specifically, each search task contains a set of connected queries.

Extracting Latent Topics After getting search tasks from the query logs, we first extract the relevant data of each search task. We use the SAT criterion to identify satisfied (SAT) clicks (as relevant data) from the query logs as either a click with dwell time of at least 30 seconds or the last click in a search session [1]. After that, we employ LDA to extract latent topics (Z) from the relevant documents of all search tasks. The LDA represents each document d as a multinomial distribution over the topics Z.

Modelling a Time-aware Search Task We model a search task with time-awareness as a multinomial distribution over topic Z as follows. Formally, we denote the

WWW 2015 Companion, May 18–22, 2015, Florence, Italy.
ACM 978-1-4503-3473-0/15/05.
http://dx.doi.org/10.1145/2740908.2742714.

search task set as S. Let s denote an instance of S. Let $D_s = \{d_1, d_2, .., d_n\}$ be a relevant document set of the search task s. We model the task s (given D_s) as a distribution over the topic Z. Furthermore, since the search task intent may change over time, the more recent relevant documents could capture more about the search intent than the distant one. This characteristic can be modelled by introducing a decay function [1]. In this paper, we model the probability of a topic z given s as a mixture of probabilities of z given relevant document $d_i \in D_s$ as follows

$$p(z|s) = \frac{1}{K} \sum\nolimits_{d_i \in D_s} \alpha^{t_{d_i}-1} p(z|d_i) \qquad (1)$$

where $\alpha^{t_{d_i}}$ is the exponential decay function of t_{d_i}; t_{d_i} is the time the searcher clicked on the document d_i within s; $t_{d_i} = n$ indicates that d_i is the n^{th} recent relevant document; and $K = \sum_{d_i} \alpha^{t_{d_i}-1}$ is a normalisation factor.

2.2 Re-ranking Search Results

For each input query q, we utilise the time-aware search task s, to which the query belongs, to re-rank the first n documents returned by a search engine. It is worth noting that s captures the topical interests of the current user.

For each returned document d, we compute a similarity measure, *TaskScore*, between d and s. Because both d and s are models as X and Y distributions over topic Z, respectively, we use Jensen-Shannon divergence to measure the similarity between the two distributions (i.e., $TaskScore = D_{JS}\lfloor X||Y \rfloor$). We consider the score as the *personalised feature*. We also extract other *non-personalised features* of q and d. Table 1 describes the features.

Table 1: Summary of the document features.

Feature	**Description**
Personalised Features	
TaskScore	The similarity score between the returned document and the search task
Non-personalised Features	
DocRank	Rank of the document on the original returned list
QuerySim	The cosine similarity score between the current query and the previous query
QueryNo	Total number of queries that have been submitted to the Search Engine

After extracting the document features, to re-rank the top n returned documents, we employ a learning to rank algorithm (i.e., LambdaMART) to train ranking models as in [1, 5].

3. EXPERIMENTS

Dataset We evaluate the approaches using the search results produced by a commercial search engine. The data used in our experiments is the query logs of 116 anonymous users in 15 days from 1st to 15th July 2012. We then partition the whole dataset into training and test sets. The training set contains the log data in the first 9 days and the test set contains the log data in the remaining days.

Experimental Methodology For evaluation, we use the SAT criterion to identify the satisfied clicks (SAT click) from the query logs. We assign a positive (relevant) label to a returned URL if it is a SAT click. Furthermore, similar to [1], we also assign a positive label to a URL if it is a SAT click in one of the repeated/modified queries in the same search session. The remaining of the top-10 URLs are assigned negative (irrelevant) labels. We use the rank positions of the positive labelled URLs as the ground truth to evaluate the search performance before and after re-ranking.

Baselines We name our proposed re-ranking model as *TimeTask*. Our first baseline, named as *Default*, is the original ranking of URLs returned by the search engine. We also construct a number of comparative baselines:

- *LongTerm*, *ShortTerm* are similar to our proposed re-ranking model. However, instead of modelling search tasks, these methods construct long-term[1] and short-term[2] user profiles respectively (see [3] for more detail).

- *StaticTask* is the non-temporal search task modelling method (that is, the decay parameter $\alpha = 1$).

Overall Performance We evaluate our proposed method by comparing the original rank list given by the commercial search engine and the re-ranked list given by our methods with four evaluation metrics: Mean Average Precision (MAP), Precision (P@k), Mean Reciprocal Rank (MRR) and Normalized Discounted Cumulative Gain (nDCG@k). For each metric, the higher value indicates the better ranking. Table 2 shows that using time-aware search tasks, TimeTask achieves better performance than the original ranking (Default) as well as other strong baselines including non-temporal search task baseline (StaticTask). It also shows that ShortTerm (with time-awareness) gains advantage over StaticTask.

Table 2: Overall performance of the methods. The improvements over *Default* are all statistically significant according to both paired t-test ($p < 0.01$) and Wilcoxon signed rank test ($p < 0.05$)

Models	Time-awareness?	MAP	P@1	MMR	nDCG@5
Default		0.6569	0.5414	0.6904	0.6853
LongTerm	✓	0.6685	0.5580	0.7057	0.7038
ShortTerm	✓	0.7217	0.6486	0.7677	0.7532
StaticTask		0.7140	0.6341	0.7581	0.7458
TimeTask	✓	**0.7304**	**0.6648**	**0.7775**	**0.7599**

4. CONCLUSIONS

We have presented a method to model time-aware search tasks using latent topics. Each search task is represented as a distribution over the topics from which we extract the personalised feature and combine it with non-personalised features to learn a ranking function using LambdaMART. We performed experiments on re-ranking search results returned by a commercial search engine. The results show that the ranking quality is improved significantly.

5. REFERENCES

[1] P. N. Bennett, R. W. White, W. Chu, S. T. Dumais, P. Bailey, F. Borisyuk, and X. Cui. Modeling the impact of short- and long-term behavior on search personalization. In *SIGIR*, pages 185–194. ACM, 2012.

[2] Z. Liao, Y. Song, L.-w. He, and Y. Huang. Evaluating the effectiveness of search task trails. In *WWW*, pages 489–498. ACM, 2012.

[3] T. Vu, A. Willis, S. N. Tran, and D. Song. Temporal latent topic user profiles for search personalisation. In *ECIR*, pages 605–616, 2015.

[4] T. T. Vu, D. Song, A. Willis, S. N. Tran, and J. Li. Improving search personalisation with dynamic group formation. In *SIGIR*, pages 951–954. ACM, 2014.

[5] R. W. White, W. Chu, A. Hassan, X. He, Y. Song, and H. Wang. Enhancing Personalized Search by Mining and Modeling Task Behavior. In *WWW*, pages 1411–1420. ACM, 2013.

[1] The user's whole search history

[2] The user's current search session