A POMDP Model for Content-Free Document Re-ranking

Sicong Zhang, Jiyun Luo, Hui Yang
Department of Computer Science
Georgetown University
37th and O Street, NW, Washington, DC, 20057
{sz303, jl1749}@georgetown.edu, huiyang@cs.georgetown.edu

ABSTRACT

Log-based document re-ranking is a special form of session search. The task re-ranks documents from Search Engine Results Page (SERP) according to the search logs, in which both the search activities from other users and personalized query log for a user are available. The purpose of re-ranking is to provide the user with a new and better ordering of the initial retrieved documents. We test the system on the WSCD 2014 dataset, in which the actual content of the queries and documents are not available due to privacy concerns. The challenge is to perform effective re-ranking purely based on user behaviors, such as clicks and query reformulations rather than document content. In this paper, we propose to model log-based document re-ranking as a Partially Observable Markov Decision Process (POMDP). Experiments on the document re-ranking task show that our approach is effective and outperforms the baseline rankings provided by a commercial search engine.

Categories and Subject Descriptors

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

Keywords

Session search; POMDP; Retrieval model

1. INTRODUCTION

The task of session search [6] is to perform document retrieval by taking in account all queries and user interactions in a session. During a session, the user writes new queries after interacting with the search engine. The interactions include clicking and reading the retrieved documents. Usually all preceding queries, previously retrieved documents, and user clicks are provided in a query log; the search engine's task is to retrieve the most relevant documents and rank them by decreasing relevance with a good use of the provided information in the log.

Recently, studies on how to utilize interaction data within a session to improve search engine's retrieval effectiveness have generated a great deal of research [1, 2, 4]. One particular research activity is the Yandex Personalized Web Search Challenge[1], which is also part of the Web Search Click Data workshop (WSCD 2014). In the WSCD 2014 task, participants are asked to re-rank the top 10 URLs from the Search Engine Results Page (SERP). Both the within-session query logs and the global query logs from the entire corpus are provided, containing document clicking information, the clicking orders, and dwell time. Specially, due to privacy concerns, there is no actual "text" provided in the dataset. Query terms are replaced by unique numeric identification numbers. Documents only show document identification numbers with no content at all (even no numeric ids for terms). For instance, in session #1690, the first query is "#3791236#2452511#2637985" and the top 3 retrieved documents are #25823561, #14641317, #23498956. We thus call this dataset a content-free document set.

SIGIR'14, July 6–July 11, 2014, Gold Coast, Australia.
Copyright 2014 ACM 978-1-4503-2034-4/14/07 ...$15.00
http://dx.doi.org/10.1145/2600428.2609529.

Such log-based content-free document re-ranking [1, 13] is a special form of session search. The re-ranking task is not to find new documents for a query but to re-order the already retrieved documents to better reflect the most current information need in the search session. The research challenge is that even with no content and no documents, how we can still be able to model the dynamics in a session and improves session search effectiveness.

In this paper we propose to model document re-ranking in sessions as a Partially Observable Markov Decision Process (POMDP) [12]. The unknown *hidden states* of the POMDP are the user's decision making states, the *belief state* is updated by user interactions in the session, and the *rewards* are IR metrics evaluating documents based on the user clicks.

Markov Decision Process (MDP) and POMDP have recently attracted attentions from IR researchers to solve a range of IR tasks [2, 4, 8, 10, 12]. While most existing work requires availability of textual content in documents, it is not a requirement in our model. As far as we know, this work is the first to apply POMDP on pure log-based document re-ranking task involving no document content.

The remainder of this paper is organized as follows. In section 2 we introduce our approach that models document re-ranking as a POMDP. Section 3 presents the experimental setup and section 4 shows the results and a discussion. Section 5 concludes the paper.

2. DOCUMENT RE-RANKING AS A POMDP

In this work, we model document re-ranking as a POMDP [5]. A POMDP is composed of agents, states, actions, rewards and transition functions. An *agent* takes inputs from the environment and outputs *actions*; the actions in turn influence the other *states* of the environment according to the *transition function.* A POMDP assigns immediate *re-*

[1] https://www.kaggle.com/c/yandex-personalized-web-search-challenge

wards for taking an action at a state. In a POMDP, the states are not known and can only be guessed through a probabilistic distribution with randomness. A set of belief states are used to model the probability that an agent is at a particular state, which are updated through receiving more observations. Below we describe the states, actions, and model optimization in POMDP for document re-ranking. Specifically, we propose to use two types of rewards during optimization; which are the *Global* reward from the entire query log from all other users and the *Local* reward based on the current session for one user only.

2.1 States

User behavior in sessions could be driven by different reasons. These can be reflected in query reformulations and clicks. Guan et al. [2] have shown how to model these user interactions using an MDP with queries as the states for session search. Here we propose two hidden decision states for session search. The two states are determined by whether the user is satisfied with the documents yielded from previous run of search or not. They are judged as *Relevant* (REL) or *Irrelevant* (IREL). Most prior research approaches user behaviors in a session with quite complex user modeling where the focus is on analysis on task types and user intent [9, 10]. POMDP could be computationally demanding if the user modeling is too complex. This research simplifies the user modeling in session search and largely reduces the scale of the state space. Hence, it is able to model document re-ranking as a POMDP with a low computational complexity.

The notations for the two states are:

- REL (Relevant): the user finds at least one relevant document from the previous SERP.
- IREL (Irrelevant): the user finds the previous SERP irrelevant.

We estimate the search relevance by the satisfactory clicks (SAT-Clicks) on the documents. A search interaction with at least one SAT-Click whose dwell time exceeds the threshold will be considered as relevant, otherwise irrelevant. We notice there are some recent novel modeling of dwell time to predict click-level satisfaction [7], however in experiment, we use a threshold of 400 time units for long dwell time, because it was given by the task as a threshold for relevance.

2.2 Actions

Generally, we define the re-ranked list of the returned documents in SERP as the action of the search engine agent. In the WSCD 2014 task, each SERP contains 10 documents to be re-ranked. Therefore, the action can be mathematically defined as an ordered vector $a = [d_1, ..., d_{10}]$, where d_i is the i^{th} document after re-ranking. In other words, each vector a stands for a new ranked list of documents.

We further group the actions into high-level strategies. These strategic rank actions are:

- **Action of Consistency.** Promote ranks for previously retrieved relevant (SAT-Clicked) documents. This action puts emphasis on the whole session consistency of the session-wise information need.
- **Action of Novelty.** Demote ranks for previously retrieved relevant (SAT-Clicked) documents. This action puts emphasis on the document novelty within a session.
- **Action of Demotion** Significantly demote ranks for previously clicked non-relevant documents. This action aims to avoid ranking the previous examined non-relevant documents high.
- **Action of No Change.** Keep the original ranks of documents.

2.3 Model Optimization

Without loss of generality, POMDP as a decision process aims to find the best decision of what action to take, based on the rewards estimation at the current state. Therefore, our model optimization is to find the optimal rank action a^* that maximizes the estimated total rewards R_{TOTAL}, which is also the total estimated document relevance:

$$a^* = \arg\max_a \{R_{TOTAL}\} \tag{1}$$

Generally, we consider the total rewards as a combination of two parts, the long-term *Global* rewards R_G and the short-term *Local* rewards R_L. Hence we re-write the optimization function as:

$$a^* = \arg\max_a \{\lambda R_G + (1-\lambda) R_L\} \tag{2}$$

The *Global* reward R_G represents the estimation of a general relevance score of the document, which can be estimated using prior knowledge across the entire log from multiple users and multiple sessions. We take the following steps to estimate the general reward of d when it is retrieved according to query q and by taking action a.

- First, we consider how much accumulated relevance score have document d gained from the training set with query q, and represent it as P(d|q). This is because the relevance score of a document is judged according to the SAT-Clicks in the dataset. Therefore we can use the likelihood of d being SAT-Clicked by other users in the training set as the estimation of the global reward.
- Second, if P(d|q) is unavailable, which implies d has not been clicked in the entire training set by any user in any session for q, we use a prior P(d) to estimate the global reward. P(d) represents the probability for d to be relevant to any query in the training dataset. In this case, although we cannot directly calculate the relevance between d and q, we use P(d) to estimate the general importance of document d, which suggests how likely d will be clicked in the entire log.
- Last, if even P(d) is unavailable, which means d has not been clicked ever in the training set. We use the original rank of document d in the current session to generate its rewards. We calculate the global reward R_G using

$$R_G = \sum_{i=1}^{10} \frac{P(d_{a,i}|q)}{log(i+1)} \tag{3}$$

where $P(d_{a,i}|q)$ stands for the rewards gained from the i^{th} document.

The *Local* rewards R_L represents the document relevance estimation based on the current session. R_L depends on believe state s and action a in the POMDP. Therefore, the reward function can be defined as $R_L = \sum_{s \in S} b(s)R(s, a)$, where s is the believe state and a is the current action. Q-learning [11] is used to calculate the local rewards. Particularly, we calculate R_L as

$$R_L = \sum_{s \in S} b(s)\delta(s) \sum_{j=1}^{n-1} \sum_{i=1}^{10} \gamma^j \frac{R(d_{a,i} \in docList_{n-j})}{\log(i+1)} \tag{4}$$

Table 1: Relevance Judgment

Score	Definition
0	(Irrelevant) Not Clicked or Clicked with dwell time less than 50 time units.
1	(Relevant) Clicked with dwell time at least 50 time units and less than 400 time units. Not include the last click.
2	(Highly Relevant) Clicked with dwell time at least 400 time units or is the last click in the search session.

Table 2: Document re-ranking example: session #9101134, where REL stands for the relevance scores of the documents according to relevance judgment ground truth.

Rank	Original List	REL	Our List	REL
1	# 8863446	0	# 8863446	0
2	# 48908975	1	# 48908975	1
3	# 48337696	0	# 48337696	0
4	# 50559948	0	# 50559948	0
5	# 14947233	1	# 14947233	1
6	# 40058654	0	# 40058654	0
7	# 45686756	0	**# 4192947**	**2**
8	**# 4192947**	**2**	# 45686756	0
9	# 3803727	1	# 3803727	1
10	# 27708292	1	# 27708292	1

where $d_{a,i}$ is the i^{th} ranked document under action a, and $docList_{n-j}$ is the previously retrieved document list in the $(n-j)^{th}$ interaction within the session. $\delta(s)$ is a parameter function which will empirically adjusts the rewards assigned to previous SAT-Clicked documents depending on the believe states. We calculate the probabilities of belief states $b(s)$ as the probabilities of the state generated from the training set. We observe that 5.7% queries lead to SATClicks. We uses SATClicked documents as relevant documents. Hence 5.7% is the maximum likelihood estimate of the belief of State "Relevant". In the similar way, we can calculate the beliefs of state "Irrelevant".

Overall, the optimization for this document re-ranking task can be concluded as :

$$a^* = \arg\max_a \{\lambda \sum_{i=1}^{10} \frac{P(d_{a,i}|q)}{log(i+1)} + (1-\lambda)(\sum_{s\in S} b(s)\delta(s) \sum_{j=1}^{n-1} \sum_{i=1}^{10} \gamma^j \frac{R(d_{a,i} \in docList_{n-j})}{\log(i+1)})\} \quad (5)$$

where $d_{a,i}$, $docList_{n-j}$, $\delta(s)$ are the same as in equation 4. γ controls the contribution ratio of the previously accumulated rewards. More details about the parameters are illustrated in section 3.

3. EXPERIMENTAL SETUP

The original dataset contains an entire search log of 27 days from an industry search engine. We separate the dataset into a training set and a test set, while we use the training data for parameter tuning and global reward calculation, and we evaluate the results on the test set. More details about the dataset are showed in Section 3.1.

Our document re-ranking experiments are performed on the test set, in which the click information according to the last SERP for each session is unavailable. The 10 urls of each of these SERPs are given as the original ranking. Our goal is to re-rank these 10 urls such that the relevant ones receive higher positions. In each search session, we use SAT-Clicks in the last SERP as *ground truth* for evaluation, while all these evaluations are based on the nDCG@10 [3] and precision@1 metrics.

Table 3: Document re-ranking example: session # 9101134. Global Rewards R_G, Local Rewards R_L & Total Rewards R_{Total} for each documents are listed.

Rank	Our List	R_G	R_L	R_{Total}
1	# 8863446	0.950	0.000	0.760
2	# 48908975	0.539	0.315	0.495
3	# 48337696	0.380	0.250	0.354
4	# 50559948	0.286	0.000	0.229
5	# 14947233	0.221	0.000	0.176
6	# 40058654	0.169	0.000	0.135
7	**# 4192947**	**0.090**	**0.315**	**0.135**
8	# 45686756	0.127	0.000	0.101
9	# 3803727	0.058	0.000	0.046
10	# 27708292	0.027	0.000	0.022

Table 1 presents the definition of the relevance judgment used for the dataset. In the dataset, the URLs (documents) are labeled using three grades of relevance. All these relevance labels are done automatically based on dwell time. It is noteworthy that, based on the user privacy consideration, all the dwell time in the dataset is presented in time units rather than actual seconds, while the data provider did not disclose how many milliseconds exactly each time unit stands for.

3.1 Dataset

Our experiments are based on an anonymized query log dataset supplied by the WSCD2014 workshop[2]. This dataset includes a one-month web search log from a well-known commercial search engine. Information available in this dataset includes user ids, queries, query terms, SERP URLs with domains, URL rankings and the click informations including dwell time. This query log is fully anonymized, therefore, only meaningless numeric IDs of users, query terms, sessions, URLs and their domains are released. The session log for the first 27 days in the dataset is fully released, hence we do the data partitioning and use the first 24 days' data as our training set to calculate global rewards, and use the following 3 days' data as test set. The following are statistics of the dataset:

- Unique queries: 21,073,569
- Unique urls: 70,348,426
- Unique users: 5,736,333
- Training sessions: 30,925,557 (from 24 days)
- Test sessions: 3,648,073 (from 3 days)
- Total queries: 65,172,853
- Total clicks: 64,693,054
- Average # of queries per session: 1.885
- Average # of click per query: 0.993

3.2 Runs Under Comparison

Baseline systems that we compare our approach with are shown in the following list.

- **Baseline-Random.** Randomly re-rank the 10 URLs in the test SERPs.
- **Baseline-SE** The original search engine returned rank list from Yandex. It is a strong baseline since Yandex is a top commercial search engine.

The runs from our approaches are:

- **Global Rewards** A document re-ranking approach only based on the global rewards as defined in Eq. 3. The local rewards estimation in POMDP are not included. This

[2] https://www.kaggle.com/c/yandex-personalized-web-search-challenge/data

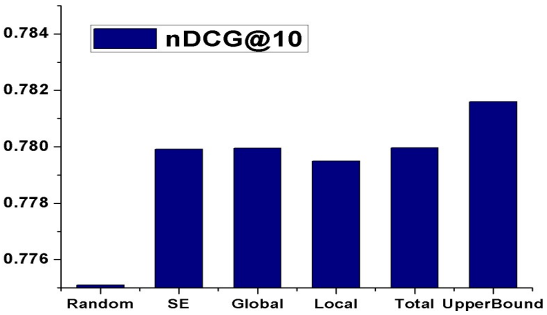

Figure 1: nDCG@10 scores for all runs

is to evaluate the effect of our Total Rewards model by comparing it with this run.

- **Local Rewards** A document re-ranking approach only based on the local rewards as defined in Eq. 4. The global rewards are not included. This is to evaluate the effect of our Total Rewards model by comparing it with this run.
- **Total Rewards** Our major run modeling the document re-ranking task as a POMDP, which considers both the global rewards and the local rewards estimations as defined in Eq. 5. The original Yandex document score is not provided in the log dataset; we hence use a document's rank in the Yandex returned rank list divided by 10 as its score. Based on the training data, the optimal weights for $P(d|q)$, $P(d)$ and Yandex document score are 1, 0.5 and 0.95. For the local reward, the optimal value of γ is 0.9.The optimal value for λ is 0.8, and the optimal value of $\delta(s)$ are $\delta(REL) = -0.5$ and $\delta(IREL) = -1$ which means at state $IREL$, a document's previous reward is treated as a stronger negative feedback to demote document rank compared with state REL.
- **Upper Bound** In this run, we use the ground truth to help choosing the strategic action with the most local rewards. Noticing this run uses the ground truth of document relevance. The goal of this run is to show the upper bound of how good this framework could achieve if we can always make the best local rewarded strategies.

4. RESULTS AND DISCUSSIONS

Table 4 presents the main evaluation results. Improvements over the original search engine ranking (Baseline-SE) are also shown. We can see our approach using $TotalRewards$ achieves improvements over the baselines in both Precision@1 (0.014%) and in nDCG@10 (0.006%). However a significant t-test(p < 0.01, one-sided) indicates that the improvement is not statistically significant.

Moreover, this run using total rewards performs better than the approaches using either only global rewards or local rewards alone. It supports our approach that models the entire rewards function for session search as a combination of both the global rewards and the local rewards. In addition, the $UpperBound$ run boosts the performance even more however with a limited amount and the improvement is statistically significant(t-test, p<0.01, one-sided). This extra improvement shows the potential ability of our framework, while on the other hand the limitation of improvements reflects the difficulty of the problem itself, especially when the baseline is the rank results from a mature commercial search engine.

Table 4: Experimental Results († indicates a significant improvement over the baseline at $p < 0.01$ (t-test, one-sided))

Approaches	P@1	Improve	nDCG@10	Improve
Baseline-Random	0.12380	−78.618%	0.46821	−39.966%
Baseline-SE	0.57900	+ 0.000%	0.77991	+ 0.000%
Global Rewards	0.57908	+ 0.014%	0.77995	+ 0.005%
Local Rewards	0.57851	− 0.085%	0.77949	− 0.054%
Total Rewards	**0.57908**	**+0.014%**	**0.77996**	**+0.006%**
Upper Bound	0.58198	+0.515%†	0.78160	+0.278%†

Table 2 and 3 shows an example in more details. Table 2 shows an example of we are re-ranking the relevant document #4192947 into a higher position, while table 3 illustrates how we achieve it by reward calculation considering both Global Rewards R_G and Local Rewards R_L.

5. CONCLUSION

In this paper, we give a POMDP approach to the document re-ranking task while the document contents are unavailable in the search log. We propose two hidden decision states REL and IREL to model user behavior, and successfully reduce POMDP computational complexity. By considering document re-ranking as a decision making process, we build up this framework handling both global rewards and local rewards to find the optimal rank action of documents, and hence to improve the re-ranking effectiveness in the testing dataset. Experiments show that our approaches outperform the strong baseline – Yandex – by 0.014%. We believe our approach gives a good novel attempt to utilize POMDP for content-free document re-ranking in sessions.

6. ACKNOWLEDGMENT

This research was supported by NSF grant CNS-1223825. Any opinions, findings, conclusions, or recommendations expressed in this paper are of the authors, and do not necessarily reflect those of the sponsor.

7. REFERENCES

[1] E. Agichtein, E. Brill, and S. Dumais. Improving web search ranking by incorporating user behavior information. In *SIGIR '06*.

[2] D. Guan, S. Zhang, and H. Yang. Utilizing query change for session search. In *SIGIR '13*.

[3] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of ir techniques. *ACM Trans. Inf. Syst.*, 20(4):422–446, Oct. 2002.

[4] X. Jin, M. Sloan, and J. Wang. Interactive exploratory search for multi page search results. In *WWW '13*.

[5] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra. Planning and acting in partially observable stochastic domains. *Artificial intelligence*, 101(1):99–134, 1998.

[6] E. Kanoulas, B. Carterette, M. Hall, P. Clough, and M. Sanderson. Overview of the trec 2013 session track. In *TREC'13*.

[7] Y. Kim, A. Hassan, R. W. White, and I. Zitouni. Modeling dwell time to predict click-level satisfaction. In *WSDM '14*.

[8] J. Luo, S. Zhang, and H. Yang. Win-win search: Dual-agent stochastic game in session search. In *SIGIR '14*.

[9] A. R. Taylor, C. Cool, N. J. Belkin, and W. J. Amadio. Relationships between categories of relevance criteria and stage in task completion. *Information Processing & Management*, 43(4), 2007.

[10] H. Wang, Y. Song, M.-W. Chang, X. He, R. W. White, and W. Chu. Learning to extract cross-session search tasks. In *WWW '13*.

[11] C. J. Watkins and P. Dayan. Q-learning. *Machine learning*, 8(3-4):279–292, 1992.

[12] S. Yuan and J. Wang. Sequential selection of correlated ads by pomdps. In *CIKM '12*.

[13] Z. Zhuang and S. Cucerzan. Re-ranking search results using query logs. In *CIKM'06*.