Exploiting Click-Through Data for Entity Retrieval

Bodo Billerbeck[†], Gianluca Demartini[‡], Claudiu S. Firan[‡], Tereza Iofciu[‡], Ralf Krestel[‡]
[†] Microsoft Research, Cambridge, United Kingdom
[‡] L3S Research Center, Hannover, Germany
bodob@microsoft.com,{demartini,firan,iofciu,krestel}@L3S.de

ABSTRACT

We present an approach for answering Entity Retrieval queries using click-through information in query log data from a commercial Web search engine. We compare results using click graphs and session graphs and present an evaluation test set making use of Wikipedia "List of" pages.

Categories and Subject Descriptors: H.3.3 [Information Search and Retrieval]: Retrieval Models. **General Terms:** Algorithms, Experimentation. **Keywords:** Entity Retrieval, Evaluation, User Session, Query Log Analysis, Wikipedia.

1. INTRODUCTION

Current Web search engines retrieve textually relevant Web *pages* for a given keyword query even if the information need targets *entities*. The idea behind *Entity Retrieval* (ER) is to find entities directly. Instead of the user browsing through all Web pages retrieved by the search engine, a list of relevant entities can be presented to the user. This not only saves the user's time but also improves search experience. Consider queries that should return a list of entities and the difficulty of compiling such a list based on keyword search results. A user looking for a list of "films shot in Venice" or "Spanish fish dishes" will have difficulties to find suitable answers. They have to manually compile the result list by extracting entities from the retrieved documents.

The idea of specifically developing a system for ER has been explored before (e.g., [2], [4]). In [1] the authors describe how to model user search behaviour exploring session data and in [5] methods are presented for named entity mining from query logs using Latent Dirichlet Allocation.

We apply the results of query log analysis to ER by performing random walks on click and session graphs. In [3] random walks are described on click graphs, containing information about clicked URLs but not about user sessions. The authors show how click graphs can be used to improve ranking of image search results. Our approach for ER extends this idea by also taking into account *session data* mined from the search query log. Thus we make use of the hidden semantic value of user session data to find relevant entities for a given ER query. We compare the usefulness of session data and click-through data for the ER task.

Copyright is held by the author/owner(s).
SIGIR'10, July 19–23, 2010, Geneva, Switzerland.
ACM 978-1-60558-896-4/10/07.

2. ENTITY SEARCH ON SESSION AND CLICK GRAPHS

Given an ER query we want to find all relevant entities and display them as a ranked list. Our hypothesis is that users posing an ER query which does not yield satisfying results will reformulate the query to find useful information. A reformulated query often consists of an instance of the group of entities the user is looking for, e.g. "Spanish fish dishes" and "Paella". This is not necessarily an ordered process but these kinds of co-occurrences can be found in user session logs nonetheless. We collect session data from a Web search engine query log and we use it to build a session graph containing each user query as a node. Two queries posed in the same 10 minute user session are connected. The direction of the edges goes from the earlier query to the more recent ones. Each of these edges is then weighted depending on the frequency of co-occurrence within different users sessions. In the second step we perform a random walk over the graph starting from a given ER query up to n steps away.

We also build a click graph, where a link between a query and a URL is established if the URL is ranked in response to the query and clicked by at least one user. Our click graph is the result of applying a Markov random walk to a large click log, as described in [3]. Similarly to the session graph, the random walk is performed for n steps away from the starting node: At search time, the given ER query is matched in the graph and set as starting node. We then perform a random walk over the graph, using query-URL-query transitions associated with weights on the edges (i.e. click frequencies) as shown in Figure 1.

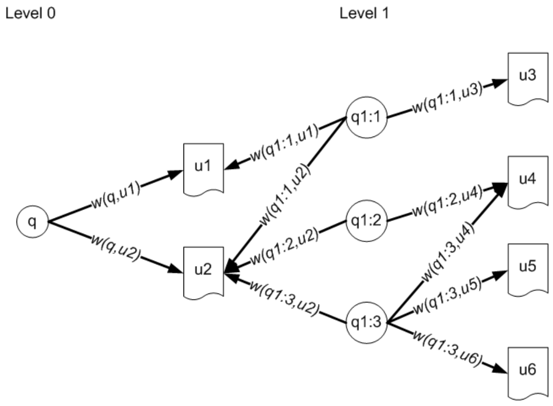

Figure 1: An example Click Graph connecting an ER query q with other queries (i.e. *entities*) $q_{l,i}$ via URLs u_l.

We compare our approaches for ranking entities using a click graph, a session graph and their intersection:

(1) RWL5 - session. Starting from an ER query, walk to all queries (interpreted as entities) reachable in 5 steps and rank all the reached queries (interpreted as entities) in the session graph by their random walk probability ([3]), but keep only queries one step away from the original query in the session graph (level 1);

(2) RWL2 - session. Similarly to $RWL5 - session$, with the random walk performed only on the first 2 levels on the session graph. That is, this approach does not explore the session graph further away than queries at level 2.

(3) RWL2 - click. Rank all the reached queries in the click graph by their random walk probability 2 steps (a step is marked for each query) away, but keep only queries closest to the original query in the click graph (i.e., one URL away from the original query);

(4) RWL2_loop - click. Similar to $RWL2 - click$, but keep only queries which can be reached by multiple paths starting from the given ER query (i.e. those that are reinforced by URLs at deeper levels) – this would keep only $q_{1:2}$ and $q_{1:3}$ in Figure 1. A level 1 query $q_{1:j}$ is considered to be reinforced as long as the path from the origin going through a different level 1 query comes back to the query $q_{1:j}$.

(5) RWL2 - intersection. After computing the intersection of the click graph and the session graph, rank all the reached queries (interpreted as entities) by their random walk probability; the random walk is performed only on the first 2 levels on each of the two graphs and only queries closest to the original one are kept.

The choice of path lengths of walks can have a large impact on the results. For this poster we experimented with path lengths of 2 and 5 steps. We plan to evaluate the effect of different path lengths more exhaustively in future.

3. EXPERIMENTS

In our experiments we use two different sets of data: (1) Query log data from the Bing search engine, and (2) queries and answers collected from Wikipedia for evaluating our approach. Both the session and click graph were built using query log data which consists of US American English language user sessions and was collected over a period of 10 months. The session graph is made up of 18 million unique queries and 65 million edges, while the click graph contains 35 million queries, 44 million URLs and 121 million edges between them.

As gold standard for the evaluation we use the "List of" pages from Wikipedia. The title of such a page, after removing the first two words, is used as an ER query (e.g., "~~List of~~ Presidents of the United States"). The titles of the Wikipedia pages that are linked to from such a "List of" page are considered to be relevant results (e.g., "Barack Obama", "George W. Bush", etc.). In order to use only queries that are more similar to typical Web queries in terms of length, we keep only those queries that consist of 2 or 3 words apart from "List of". Thus we have 17,110 pages out of the total of 46,867 non-redirect "List of" pages. We match these titles to queries in the log (exact string match) and keep only the 82 queries which were posed at least 100 times and attracted at least 5 clicks on results[1]. In order to compare the different ranking approaches, we compute MAP and R-Precision of the produced rankings. For the purpose of evaluation we stem both the retrieved queries and the relevant results. Furthermore, we consider the first ER query in each ranked list as relevant if it contains (as a substring) any entry in the respective "List of" page. Because of this, the paper at hand should be viewed as a head-room experiment; in future work we plan to extract entities from queries before matching these to the target set for the purpose of evaluating our approach. As a side note, according to the definitions above, the session and click graphs used for this experiment cover roughly 60% and 75% of relevant entities, respectively.

Method	MAP	P10	Retrieved
$RWL5 - session$	0.2372^*	0.1175	60
$RWL2 - session$	0.2450^*	0.1173^*	61
$RWL2 - click$	0.1862^+	0.0483^+	531
$RWL2\_loop - click$	0.1911^{*+}	0.0545^+	399
$RWL2 - intersection$	0.3462^{*+}	0.1918^{*+}	21

Table 1: Results for finding entities using click and session graphs. * indicates statistical significant difference with $RWL2 - click$, + with $RWL2 - session$ (paired t-test, $p < 0.05$).

4. DISCUSSION AND CONCLUSIONS

We can see in Table 1 that the approach based on reinforcement improves over a standard random walk in the click graph. Performing a random walk and ranking the queries by their respective probabilities works better using the session graph than the click graph. This can be explained because users typically start a search session by posting a generic query such as "Spanish fish dish" and refining it with "Spanish fish dish paella". Additionally, we can see that walking only two levels yields even better performances (as most of the relevant results are one step away from the original ER query) which is also computationally more efficient. Interestingly, using the session graph retrieves *overall less* entities but *more relevant* ones per query. This shows how using the session graph is a more suited approach as the average Web user would not browse hundreds of results. Finally, we can see that when computing the intersection of the two graphs we obtain best effectiveness. This means that results contained in both graphs are mainly relevant ones. Moreover, the number of retrieved results is reduced to a minimum which is realistic for the average Web user. This proves that large query logs from Web search engines can be successfully used for the emerging task of Entity Retrieval. A combination of the methods presented here with traditional IR is definitely worth investigating.

5. REFERENCES

[1] Ricardo Baeza-Yates and Alessandro Tiberi. Extracting semantic relations from query logs. In *KDD*, 2007.

[2] Tao Cheng, Xifeng Yan, and Kevin Chen-Chuan Chang. Entityrank: searching entities directly and holistically. In *VLDB*, 2007.

[3] Nick Craswell and Martin Szummer. Random walks on the click graph. In *SIGIR*, 2007.

[4] Desislava Petkova and W. Bruce Croft. Proximity-based document representation for named entity retrieval. In *CIKM*, 2007.

[5] Gu Xu, Shuang-Hong Yang, and Hang Li. Named entity mining from click-through data using weakly supervised latent dirichlet allocation. In *KDD*, 2009.

[1] The test set of Wikipedia titles and relevant entities is available from `http://www.l3s.de/~iofciu/wikipediaER/`