Query Recommendation in the Information Domain of Children

Sergio Duarte Torres DUARTES@UTWENTE.NL

PO Box 217, Database Group, University of Twente, Enschede, The Netherlands

Djoerd Hiemstra HIEMSTRA@UTWENTE.NL

PO Box 217, Database Group, University of Twente, Enschede, The Netherlands

Ingmar Weber IWEBER@QF.ORG.QA

Qatar Computing Research Institute, Doha, Qatar

Pavel Serdyukov PAVSER@YANDEX-TEAM.RU

Leo Tolstoy st. 16, Yandex, Moscow, Russia

Abstract

Children represent an increasing part of web users. One of the key problems that hamper their search experience is their limited vocabulary, their difficulty to use the right keywords, and the inappropriateness of general-purpose query suggestions. In this work we propose a method that utilizes tags from social media to suggest queries related to children topics. Concretely we propose a simple yet effective approach to bias a random walk defined on a bipartite graph of web resources and tags through keywords that are more commonly used to describe resources for children. We evaluate our method using a large query log sample of queries submitted by children. We show that our method outperforms by a large margin the query suggestions of modern search engines and state-of-the art query suggestions based on random walks. We improve further the quality of the ranking by combining the score of the random walk with topical and language modeling features to emphasize even more the child-related aspects of the query suggestions.

1. Introduction

Children experience several difficulties searching the web using state-of-the-art search engines. In particular, children have been found to struggle formulating keyword queries, exploring the list of web results and finding relevant information (Bilal, 2002; Druin et al., 2009). Moreover, children may be exposed to inappropriate material given the lack of constant parental supervision and the vast amount of information online that is not suitable for them.

In this work we propose a query suggestion method to help children find keywords that are more likely to be

This work was partially carried out while the first and third author were at Yahoo Labs! Barcelona as an intern and research scientist, respectively.

relevant for them. Query suggestions alleviate the problem of finding the right keywords to search the web, which is particularly challenging for children (Druin et al., 2009). Extensive research has been carried out on general-purpose query suggestion (Wang and Zhai, 2008; Baeza-Yates and Tiberi, 2007). This work deviates from previous studies in that *(1)* the suggestions are aimed at children's search intents; *(2)* the suggestions are constructed in the absence of query logs; *(3)* the suggestions are ranked based on a novel biased random walk to promote suggestions aimed at child-specific topics and that *(4)* we combine the results of our random walk in a learning to rank approach with topic and language modeling features to focus the ranking on more children friendly suggestions.

The query suggestions of our method are based on the tags from the bookmarking system *Delicious* that are associated to the query web results and to previously seen web resources intended for children. We consider tags a reasonable source of terms for query expansion given the high overlap of tags and query terms found in large query logs (Yanbe et al., 2007). These type of tags are a valuable resource in the IR domain for children since we can exploit the collaborative information provided by users sharing web resources for children. The method proposed in this work also mitigates the problem of finding irrelevant and unsuitable information since our suggestions boost search aspects that are related to children search intents.

Concretely, we propose a novel way to boost tags in a random walk that are frequently used to describe resources for children and that are prominent with respect to a background model of web resources aimed at the general public. The model is built using a set of bookmarks of high quality web resources focused on content for children. The assumption of our method is that tags frequently associated to urls focused on children topics are better candidates to construct query suggestions for children. For instance consider the query *cars*. According to Google's query suggestions common aspects of this query are *car rentals, cars for sale, used cars, new cars, disney cars and car pictures*. On the other hand, aspects oriented to satisfy children information needs should rather include *car movies, car games, car toys, car coloring pages, car pictures, car crafts*. Our system ranks higher the latter tags, providing more focused suggestions on content for children.

We refine the ranking of the query suggestions obtained by the random walk using a learning to rank approach. We employ the random walk score along language modeling and topic features based on the category structure of the Dmoz *kids and teens* section, in this way we emphasize further the suitability of the suggestions for children and the system is informed about the relevant topics associated to the query. It is important to mention that our query expansion method is complementary to recent approaches found in the literature to improve the search experience of children by filtering inappropriate content (Eickhoff et al., 2010) and other search functionality such as AgeRank (Gyllstrom and Moens, 2010).

The quality of the results is evaluated using a large anonymized log sample of queries and query reformulations from users aged 8 to 18 years old extracted from the Yahoo! Search engine. We show that the results also hold when using the AOL query log with query reformulations of queries used to retrieve content aimed at children.

The organization of this paper is as follows: Section 2 discusses the relevant related work to this paper. Section 3 describes our method. Section 4 describes the data acquisition process. Section 5 presents the results obtained by our random walk method. Section 6 presents the features utilized to improve further the ranking of results and the experimental results. In the last section conclusions of this work and some directions for future work are discussed.

2. Related work

This work is related to the areas of query suggestion, query expansion, tag ranking and biased random walks.

2.1. Query Recommendation

Extensive research has been carried out on query recommendation based on query click-through data from query logs (Gao et al., 2010; Baeza-Yates and Tiberi, 2007). In these methods the association between query and documents in the search graph is mined to infer related queries. More recently, random walk frameworks have been proposed to rank documents and queries using hitting time (Mei et al., 2008) and based on the query - document frequency in the graph (Boldi et al., 2009; Craswell and Szummer, 2007).

The method we propose utilizes Craswell and Szummer (2007) framework. However, our work deviates from theirs in the definition of the transition probabilities and the normalization of these probabilities. Our motivation is to bias the walk towards suggestions more appropriate for a specific niche of users (i.e. children) which is not addressed by their work.

Recently, two random walk frameworks have been proposed to leverage query log and social media annotation within the same graph. The first exploits the latent topic space of the graph (Bing et al., 2011) and the second utilizes the graph hitting time (Mei et al., 2008). Their focus is to refine queries by exploiting the tag vocabulary of the social media and to provide exploratory and search query suggestions within the same framework. Our work also exploits the annotations of social media for the generation of query suggestions. However, our work differs in that the query suggestions are generated in the absence of query logs, that is solely based on social media. Their work also does not address the generation of suggestions for a specific group of users, which we address by introducing a novel bias into the random walk. Finally, our work differs in that we integrate the scores of the random walk in a learning to rank approach to emphasize further the suitability of the suggestions for children.

2.2. IR for Children

Duarte Torres et al. (2012) presented a biased random walk to recommend tags as query suggestions for children between 10 to 12 years old. Partial results were shown using the AOL query log. In this work we greatly improve and extend this random walk model by using backward propagation probabilities and by using a learning to rank approach in which the random walk score is combined with novel topical, language modeling and tag similarity features. We also show that our method is suitable for other age groups (e.g. teenagers) by evaluating the proposed method on a set of queries submitted by actual young users.

Gyllstrom and Moens (2010) presented a variation of Page Rank to rank web pages that are more suitable for children. They utilized label propagation to score documents. Our work deviates in the characteristics of the graph utilized (we employ social media while they employ web documents) and in that we boost query suggestions associated with content for children through information metrics between a foreground model of tags used to describe content for children and a background model. Moreover we improve over the results of our random walk using a learning to rank framework by introducing features that are not trivial to add in a graph based model. Gyllstrom and Moens (2011) presented a method to detect queries that represent controversial topics and topics for children. Their method filter topics by relying on the query suggestions of a commercial search engine. In our work we provide query suggestions in the absence of search engine's query suggestion functionality. Eickhoff et al. (2010) proposed a machine learning approach to filter content unsuitable for children. Although both problems are different (they addressed a binary classification problem), some of the features we use are similar to the ones used in their research. Nonetheless, the main feature in our approach is the random walk proposed.

It is also worth to mention other efforts from the PuppyIR European project in which some the content filtering, query suggestion, and topic detection mentioned has been applied and presented as showcases (Azzopardi et al. (2012a;b;c))

2.3. Tag Ranking

Tag ranking has recently received attention given the proliferation of social media sharing sites. Liu et al. (2009) proposed a method to estimate the relevance score of a tag to an image based on probability density estimation. The estimation is further refined using a random walk over a tag similarity graph. Several tag ranking methods have recently been proposed in the domain of Image tagging (Li et al., 2008; 2012; Zhuang and Hoi, 2011; Feng et al., 2012; Zhu et al., 2010). On overall, these methods exploit the similarity between tags based on the neighboring tags in the graph. Additional evidence such as visual and semantic features and are also employed to reduce noise and disambiguate the meaning of the tags.

Our work deviates in the structure of the graph and the bias introduced into the random walk. In the previous works mentioned the graph consists only of tags and in our problem it consists of tags and web resources. This graph structure is important since we exploit the characteristics of web resources aimed at children to bias the

`http://hmi.ewi.utwente.nl/puppyir/`

random walk. Moreover, they do not consider the age dimension of the users.

2.4. Biased random walks

Biased random walks have been proposed before in different problem domains. Haveliwala (2002; 2003) proposed the Topical Page Rank. This is a variation of PageRank in which the topic of the query and the urls are taken into account. The PageRank score is refined by calculating multiple scores for each page, each score representing the importance of the page in respect to a topic. These scores are then combined according the importance of each query for the topic. The performance of this method is further explored by Kohlschütter et al. (2007). They found that the performance of this method varies according the specificity of the topics considered, and that more specific topics tend to lead to better results. Qiu and Cho (2006) build on the Topical Page Rank and includes the user clicking story to enhance the ranking. Abou-Assaleh et al. presents a similar method to Topical Page Rank, in which the search is focused on specific topics. They obtained comparable results with less computational overhead.

Wu and Chellapilla (2007) proposed a biased random walk to extract link spam communities when at least one of the members of this community is known. Their method is referred as Spam Rank. The bias is introduced by employing decay probabilities in which nodes having a greater distance from the seed set are penalized. Spam detection can be mapped to our problem by seeing the seed set as the set of keywords of interests for young users Zhang et al. (2009) expanded on Wu and Chellapilla (2007)'s work by proposing a method to automatically enlarge the seed set. Gyöngyi et al. (2004) addressed the problem of ranking pages avoiding spam. They used a seed set of trusted resources to avoid spam instead of having a seed set to detect spam communities.

Fuxman et al. (2008) presented a random walk with absorbing states for the generation of keywords in the domain of sponsored search. The random walk is defined on a bipartite graph of queries and urls constructed from query logs. A set of seed queries or urls are set as absorbing states to bias the random walk towards these states and the relationship between queries and urls in the graph are exploited to generate keyword suggestions. Their approach relates to our problem in that we are interested in biased the random walk from a seed set of urls and tags. The generation of keywords in this domain has recently been explored further (Ravi et al., 2010; Hui et al., 2013).

In section 4 we present a more detailed description of three methods: Topical Page Rank, Spam Rank and the generation of keywords proposed by Fuxman et al. (2008). We describe details on how these methods can be mapped to the scenario addressed in this paper.

3. Method

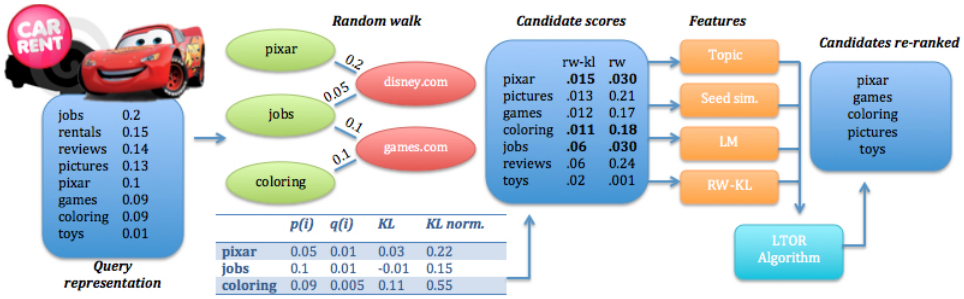

Figure 1. Query Suggestions Framework.

We envisage a search service for children which uses state-of-the art search engines to deliver content aimed at children. In this system, the query submitted by the user is sent to several search engines to retrieve keywords from the snippets and titles of the web results. These keywords represent the possible topics associated to the user's query. Our task is to generate these keywords and rank them to construct query suggestions. The ranking is carried out by first generating a ranked list of candidate suggestions using the random walk. Secondly, the candidates ranking is refined by using a learning to rank approach in which the random walk score is combined

with topical and language modeling features. Figure 1 depicts the framework employed to rank query suggestions. Note that under this scenario we do not have access to search engine query logs which are widely used for query recommendation (Boldi et al., 2009; Ma et al., 2011). Although in our architecture it would be possible to register log activity, we would still face the cold start problems of not having data to generate the query suggestions. Moreover given increasing privacy concerns and the characteristics of the audience targeted by our system (i.e. children), it is desirable to avoid gathering user information. Recent search engines as *DuckDuckGo* and *Yippy* are gaining popularity in part for their policy of not storing user data.

3.1. Random Walk Towards Content for Children

Our random walk model uses a bipartite graph of web resources (i.e. *urls*) and tag nodes. Previous research on tag ranking (Liu et al., 2009) employed random walks methods for tag recommendation systems using a graph composed solely of tags. In the setting of our problem we found it useful to treat urls as nodes as well since our methods rely on a trusted set of web resources, which are used as seeds to bias the random walk towards more relevant tags for the targeted audience. That is, tags more frequently associated to urls that are known to be targeted at certain niche of users (i.e. children) will be promoted over tags employed more frequently to described urls for a different niche of users (i.e. adults). Note that it is not straightforward to represent this information in the case where the graph is only composed of tag nodes, moreover this graph representation allows to add a measure of how reliable or trustful a seed url is.

In this work the graph was created using a set of the *Del.icio.us* bookmarks from the collection presented by Wetzker et al. (2008). Concretely, bookmarks of urls known to be adequate for children and young audiences were extracted to create the set of urls and tags. Details about the characteristics of the dataset are provided in Section 5.1. Our random walk method is based on the framework proposed by Craswell and Szummer (2007). Formally the graph is defined as:

Definition 1 *a* (bipartite graph of urls and tags)*:* $G = (U, T, E = \{(u,t)|(u,t) \in U \times T\})$ *where* $U = \{u_1, u_2, .., u_n\}$ *is the set of urls described by tags* $T = \{t_1, t_2, .., t_m\}$ *and* E *is the set of edges in the graph.*

Craswell and Szummer (2007) defines the transition probabilities as:

$$p_{fw}(i|j) = \begin{cases} (1-\alpha)\frac{c(i,j)}{\sum_{k:(j,k)\in E} c(j,k)} & if\ i \neq j \\ \alpha & if\ i = j \end{cases} \quad (1)$$

$$p_{bw}(i|j) = \begin{cases} (1-\alpha)\frac{p_{fw}(j|i)}{\sum_{k:(i,k)\in E} p_{fw}(k|i)} & if\ i \neq j \\ \alpha & if\ i = j \end{cases} \quad (2)$$

For the cases of forward and backward random walk respectively. The term $c(i,j)$ represents the number of times a tag i was used to describe a web resource j and the term α is the self transition probability which is used to slow the diffusion of the scores. We employed both types of weight functions as baselines. As it was observed by Boldi et al. (2009), we found that the backward weight performs better, although only marginally for our problem, as will be shown in the next section.

We propose to bias the random walk by introducing a weight based on the point-wise Kullback-Leibler (KL) divergence metric. Intuitively, this metric allows to promote those tags that have a larger probability to appear in a collection of content for children (our foreground model) than in a corpus of content for adults (background model). This intuition is exemplified in Figure 1 using the query *cars*. In this example the most popular keywords associated to the query are *jobs*, *rentals* and *reviews*. Using the baseline random walk we obtain *pixar*, *jobs* and *reviews* as the top ranked results. However, when the *KL* weight is introduced, the latter two keywords are penalized further allowing keywords such as *pictures*, *games* and *coloring* being better ranked. Equation 3 and 4 reflect the new transition functions.

https://duckduckgo.com/privacy.htm
http://search.yippy.com/privacy

$$p_{fwKL}(i|j) = p(i)\log\frac{p(j)}{g(j)}p_{fw}(i|j) \tag{3}$$

$$p_{bwKL}(i|j) = \begin{Bmatrix} (1-\alpha)\frac{p_{fwKL}(j|i)}{\sum_{k:(i,k)\in E} p_{fwKL}(k|i)} & if\ i \neq j \\ \alpha & if\ i = j \end{Bmatrix} \tag{4}$$

where $p(i)$ is the probability of a tag (or url) to appear in the collection of resources for children and $g(j)$ is the probability of i to appear in the collection of resources for the general public. We normalize the point-wise Kullback-Leibler (KL) distances to lie between 0 and 1 in order to introduce them into the random walk framework. The normalization was carried out using the maximum and minimum KL point-wise distance in the collection in the following manner: $kl_n(p||q) = (kl(p||q) - \min(KL))/(\max(KL) - \min(KL))$.

We also found that using a uniform normalization for the transition of *urls* to *tags* improves the performance of the random walk. Intuitively, this occurs because the standard transitions of urls to tags tend to promote the most popular tags. However, our focus is to promote tags that are more children oriented, which are not necessarily the most popular for a given url. Thus, a uniform normalization emphasizes the effect of the KL weight introduced in Equation 3 and 4. Using this observation we renormalized the forward probability as follows:

$$p_{fwN}(i|j) = \begin{Bmatrix} (1-\alpha)\frac{c(i,j)}{\sum_{k:(j,k)\in E} c(j,k)} & if\ i \neq j,\ j \in T \\ (1-\alpha)\frac{p_{fw}(j|i)}{\sum_{i:(j,i)\in E} p_{fw}(j|i)} & if\ i \neq j,\ j \in U \\ \alpha & if\ i = j \end{Bmatrix} \tag{5}$$

From Equation 3 we need to estimate the probabilities of the tags and urls in the two corpora. These probabilities are estimated based on a set of Delicious bookmarks that represent the interests of the target group.

We define a bookmark as a tuple containing a url and a tag which describes the url: $b = \langle u_i, t_i \rangle$ where $u_i \in U, t_i \in T$, the set of urls and tags respectively. A collection of bookmarks is defined as a bag of N bookmarks $B = \{1, b_2, .., b_N\}$.

We employ a set of bookmarks that contains trusted urls oriented towards a specific audience: children and teenagers.

Definition 2 *The* (bag of bookmarks of trusted and oriented urls for a target audience) *is defined as:* $B_k = \{b_1, b_2, .., b_N | url(b_i) \in U_k\}$ *where* U_k *is the set of seeds urls and* $url(b_i)$ *extracts the url from the bookmark* b_i*.*

The estimation of the transition probabilities depicted in Equation 3 is estimated using maximum-likelihood estimation (MLE) using B_k for the foreground model and B for the background model.

$$\begin{aligned} p(t) = \frac{cf_{B_k}(t)}{|T|}, p(u) = \frac{cf_{B_k}(u)}{|U|} \\ g(t) = \frac{cf_{B}(t)}{|T|}, g(u) = \frac{cf_{B}(u)}{|U|} \end{aligned} \tag{6}$$

where $|T|$ and $|U|$ are the raw size of tags and urls in the collection B_k.

3.2. Query Representation

The query is represented as a single node in the graph and we define a special transition probability from the query node to the tag nodes of the graph. We do not include transition probabilities from the query to url nodes because the user's query is represented as a bag of tags. The query representation is constructed from the query itself and the tags found in the titles and snippets of the top ranked web results. The query can also be seen as a document constructed with the tags found in the web results and the query. Formally we define the user's query and the tag set of a query as:

Definition 3 (Query) *A query* q *of length* l *is represented as the sequence of words* $(w_1, w_2, .., w_l)$*.*

Definition 4 (Tag set of a query) *The tag set of a query q consists of the m tags extracted from a social bookmarking system S, which are associated to the top web results of query q:* $Q = \{t_1, t_2, .., t_m\}$.

This representation is convenient because query suggestions can often be obtained directly from the keywords appearing in the snippets of the web results (Yanbe et al., 2007). Using this query representation we define the transition probability $p(t|Q)$ as:

$$\begin{aligned} p(t|Q) &= \frac{p(Q|t)p(t)}{p(Q)} \\ p(t|Q) &\propto p(t)p(Q|t) \\ p(t|Q) &\propto p(t)\prod_{i=1}^{|Q|} p(q_i|t) \end{aligned} \tag{7}$$

The first term on the right hand side is the likelihood of the candidate tag t in the collection and the second term describes the likelihood of t co-occurring between the tags in the query and the collection. These probabilities are estimated using MLE in a similar fashion as in 6

$$p(q_i|t) = \frac{cf(q_i, t) + \mu\, p(q_i)}{|T| + \mu} \tag{8}$$

where $p(q_i)$ is the prior probability of q_i and μ is the Dirichtlet smoothing parameter.

4. Related biased random walks

In the following paragraphs we present the description of other biased random walks that can be applied to our problem. We point out the differences of these methods with our approach and we provide relevant implementation details adopted for their comparison against our method.

4.1. Topic-sensitive page rank

The topic-sensitive page rank proposed by Haveliwala (2002) builds on the definition of PageRank, in which the scores of the nodes are computed in the following manner:

$$\vec{Score}_{t+1} = (1-\alpha)M \times \vec{Score}_t + \alpha\vec{p} \tag{9}$$

where the element m_ij of matrix M is equal to $1/N_j$ (the inverse of the number of outgoing links of node j) if there is a link from node j to node i and 0 otherwise. And $\vec{p} = \left[\frac{1}{|N|}\right]_{N\times 1}$.

Thus, the prior probabilities are uniform for all the nodes and the transition probabilities are normalized uniformly by the number of outgoing edges. The bias is introduced by using as vector $\vec{p}$ the topic vector $\vec{v}$ (i.e. $\vec{p}$= $\vec{v}$), in which each element in this new vector is defined in the following manner:

$$v_i = \begin{cases} \frac{1}{|T_j|} & i \in T \\ 0 & i \notin T \end{cases} \tag{10}$$

Haveliwala (2002) defines the query representation based on the probability of the query given the target topic:

$$p(c_j|q) = \frac{p(c_j)p(q|c_j)}{p(q)} \propto p(c_j)\prod_i p(q_i|c_j) \tag{11}$$

where $p(c_j)$ is the probability of topic c_j (set uniformly in (Haveliwala, 2002)) and $p(q_i|c_j)$ is estimated using MLE based on the documents of the Dmoz category c_j. The scores of all the topics are combined in the following manner:

$$s_{qn} = \sum_j p(c_j|q)score_{jn} \tag{12}$$

where $score_{qn}$ is the score of node n in the graph (e.g. a url in PageRank) for the query q.

The first key difference of the topic-sensitive PageRank method with the method proposed in our work is that all the nodes are considered of the same class, while we propose a graph that distinguishes between tag and url nodes. Also note that the main bias introduced by this method arises in the non-uniform probabilities in vector $\vec{v}$, which makes uses of the number of outgoing links belonging to the target topic. This bias is implicit in our method by the construction of the graph in which only urls and tags from trusted resources for children are utilized. We implement the topic-sensitive PageRank by considering that we are only targeting topics for children (or teenagers). Thus we do not employ a vector of scores of different topics, instead we estimate the scores for the children set of urls found in Dmoz as we do for our method. We also utilize the query definition proposed in Equation 8 to make the results comparable to our method.

4.2. Seed based random walk

Fuxman et al. (2008) proposed a random walk using a bipartite graph of urls and queries (tags in our problem scenario) from query logs. Their algorithm assumes as input the graph, a set of concepts and a set of seeds of urls or tags in which a seed is mapped to a specific class. Their task is to recommend urls or tags related to the seed set representing the concepts. We map their model to our settings by employing only one class (*i.e.* content for children) and by employing the bag of tags that represent the user's query as the set of seeds. The task in our problem is to find other tags related to the seed tags. The transition probabilities of this random walk are defined in the following fashion (Fuxman et al., 2008):

$$p(l_t = c) = (1 - \alpha) \sum_{u:(t,u)\in E} w_{tu} \cdot p(l_u = c) \tag{13}$$

$$p(l_u = c) = (1 - \alpha) \sum_{t:(u,t)\in E} w_{ut} \cdot p(l_t = c) \tag{14}$$

where the weight values $w_{tu} = \frac{c(t,u)}{\sum_{k:(t,k)\in E} c(t,u)}$ and $w_{ut} = \frac{c(u,t)}{\sum_{k:(u,k)\in E} c(u,k)}$ are the transition probabilities from node t to node u and vice-versa. The value $p(l_t = c)$ represents the probability of a tag or url of being absorbed by the nodes in the set seed during the random walk. In practice these values are the accumulated random walk score.

Note that these transition probabilities are equivalent to the transition probabilities defined in Equation 1. The main difference of the method proposed by Fuxman et al. (2008) and our method is the way the bias is introduced to the random walk. Fuxman et al. (2008) introduces the bias by establishing all the nodes from the seed set (either urls or tags) as absorbing states. This is accomplished by setting $p(l_t) = 1$ if the node t belongs to the seed set and $p(l_t) = 0$ otherwise. Note that this implies that the values of the nodes belonging to the seed set are not updated with the random walk since they are set to 1 in the initialization step. Another difference is the special normalization scheme adopted for the random walk (Equation 5) and the use of a special query node to represent the query tags. Additionally, a threshold is employed by Fuxman et al. (2008) to set as null absorbing states (nodes with $p(l_t) = 0$) those nodes in which the accumulated probability fall below the threshold. This threshold is also used for efficiency purposes in the implementation.

4.3. Spam detection random walk

Wu and Chellapilla (2007) employed a biased random walk to extract spam communities. The input of the algorithm is a graph (all the nodes are of the same type) and a seed set of nodes, which are used to bias the random walk. We map this method to our problem scenario by employing the query bag of tags as the seed set. In this case the tags related to the seed tags are seen as the spam community to be detected.

The node probabilities are updated using the following expression:

$$p(i)_{t+1} = \frac{1}{2} \left[I + AD \right] p_t(i) \tag{15}$$

where I is the identity matrix, A is the adjacency matrix of the graph and D is the diagonal matrix in which

the elements $d_{ii} = \frac{1}{d(i)}$ where $d(i)$ is the degree of node i. The bias is introduced in the initialization step of the random walk, by setting the node probabilities in the following fashion:

$$p(i)_t = \begin{cases} 1/|S| & if\ i \in S \\ 0 & otherwise \end{cases} \tag{16}$$

where S is set of seed nodes and $|S|$ is the size of the set.

Wu and Chellapilla (2007) added four additional constraints to the random walk in the implementation: *(i)* The probability scores are truncated to zero if they fall under a specified threshold or if they fall in the bottom of the k-percentile probability distribution. We adopted the former in our implementation. Note that an analogous parameter is utilized in Fuxman et al. (2008) for optimization purposes; *(ii)* the probabilities are renormalized to sum to one after each random walk iteration. This process is carried out if there are leaf nodes with no children nodes. However, this situation does not occur given the construction of our bipartite graph; *(iii)* a list of trusted (*white*) domains are considered to avoid the random walk to follow links to these set of domains. This restriction is reasonable in the problem of spam detection since trusted domains are unlikely to link to spam. However, we believe this restriction is specific for the problem of spam detection and we did not considered it in our problem scenario; and *(iv)* the node probabilities are biased by penalizing those nodes that have a greater distance in respect to the seed nodes. This is carried out by weighting the node probability $p(i)_t = p(i)_t \cdot 2^{\delta}(i)$, where $\delta(i)$ is the shortest path distance of node i to the nodes in the seed set. This restriction was considered.

This method differs from our method in that it does not distinguish the types of nodes (as it was the case with the topic-sensitive PageRank) and the transition probabilities are defined based on the number of nodes on the seed set, thus the information about the relationship of tags and urls is not captured by this method. As it was the case with the previous methods, the information of suitability for children represented through the point-wise Kullback Leibler divergence metric is not captured.

5. Data set extraction

5.1. Training Data

As training data we created a set of *Del.icio.us* bookmarks from the collection created by Wetzker et al. (2008). To the best of our knowledge this is the largest collection of social tagged data available for research. The collection contains 132 million bookmarks and 420 million tag assignments and was retrieved between December 2007 and April 2008. The set was created by extracting the bookmarks of the urls listed in the *Kids and Teens* section of *Dmoz*. These urls link to "web sites that have been selected for age-appropriate content by a team of volunteer editors"[1]. These resources have also been used in other information retrieval problems for young users (Gyllstrom and Moens, 2010; Eickhoff et al., 2010) with positive results.

The data cleaning process has a particular importance in our problem since we require well-formed and meaningful tags to construct query suggestions. We observed that tags are noisy and their usage varies greatly among users. We estimated that 9% of the tag volume were not useful as candidates for query expansion, either because the tags were not descriptive or because they refer to web addresses (e.g. *to-see, www.sfgate.com*). We also found a high percentage of ill-formed descriptive tags (e.g. *artist (music)*) and a large percentage of multi-worded tags: both with and without token separators (e.g. *new-york, avrillavigne*). Traditionally these problems are addressed relying on the redundancy of the data. However, in our problem other strategies are required given that the volume of information aimed at children (and in general to a niche of users) is significantly smaller than the volume of data aimed at the average user.

The data cleaning process was carried out in two steps: *tag normalization* and *tag filtering*. For the normalization we first follow a rule-based approach to generate a homogeneous representation of the multi-worded tags. Token separators such as "_", ".", " " were normalized to the character "_". For example this procedure maps the tags *star.wars* and *star-stars* to *star_wars* To normalize multi-worded tags (e.g. *avrillavigne*) we define a relation R between the set of tags without token separation T and the set of known multi-worded tags in the collection MT, in which each tag of the form $x_1x_2..x_n$ is associated to one or more of their split forms (if any) $x_1\_x_2\_..\_x_n$.

[1] http://www.dmoz.com

The relation is defined as

$$R = \{(a, B) | a \in T, B \subseteq T_h, b \in T_h \wedge a = b_{untokenized}\} \tag{17}$$

where $b_{untokenized}$ is equal to the tag b without any token separation. This relation gives us a set of candidate split forms for a target tag. However, it is still necessary to decide when the tag has to be split since the split form of a tag is not always the correct form (it may be due to misuse language). Three features were employed to decide when tags of the form x_1x_2 should be split into $x_1\_x_2$: *(1)* normalized point-wise mutual information (*nPMI*); *(2)* the ratio between the frequency of the tag in the form $x_1\_x_2$ and the form x_1x_2 ($\frac{f_{x_1\_x_2}}{f_{x_1x_2}}$); *(3)* and the frequency of the tag in the form $x_1\_x_2$.

PMI is commonly used in NLP for mining collocations. A drawback of *PMI* is its sensitivity to the frequency of the terms involved in the calculation (Van de Cruys, 2011). Higher *PMI* values indicate a higher association between the terms, nonetheless high values can also hold even if the two terms rarely occur in the collection. For this reason we employed *nPMI* which is less sensitive to the sparsity of the data. Equation 18 shows how the *nPMI* is calculated for two terms. To calculate the *nPMI* for n terms we employed the total correlation information metric, which is one of the possible generalization of *PMI*. This metric measures the amount of information that is shared among a set a random variables (Van de Cruys, 2011). Equation 18 and 19 shows its definition.

$$\begin{aligned} pmi(x_1, x_2) &= \log\left[\frac{p(x_1, x_2)}{p(x_1)p(x_2)}\right], \\ npmi(x_1, x_2) &= \frac{pmi(x_1, x_2)}{-\log\left[\max(p(x_1), p(x_2))\right]} \end{aligned} \tag{18}$$

$$\begin{aligned} pmi(x_1, x_2, .., x_n) &= \log\left[\frac{p(x_1, x_2, .., x_n)}{\prod_{i=1}^{n} p(x_i)}\right], \\ npmi(x_1, x_2, .., x_n) &= \frac{pmi(x_1, x_2, .., x_n)}{-\log\left[\max(p(x_1), p(x_2), .., p(x_n))\right]} \end{aligned} \tag{19}$$

The ratio $\frac{f_{x_1\_x_2}}{f_{x_1x_2}}$ was employed as a feature to decide when to split hyphenized tokens since in some cases the terms within a split tag have a high level of association but the correct form is as a single token (e.g. hummingbird). The threshold values for the three features were set experimentally to maximize precision on a sample of 2,000 tags without being split and their manually annotated correct split form. Concretely, by setting the parameter *(i)npmi* to 0.4, *(2)* the ratio to 0.015 and the *(3)* frequency threshold to 3 we were able to obtain a maximum precision of 87% on the sample extracted.

We also filter out tags satisfying any of the following conditions: *(i)* is in the dictionary of tags that refer to adult or explicit content; *(ii)* is used for personal administrative purposes (i.e. *to-do, to-see*); *(iii)* contains non-alphanumerical characters; *(iv)* is a url or points to a web service or *(iv)* was submitted by less than 3 users in the entire collection.

5.2. Test Data

We employed a large anonymized sample of search logs from the Yahoo! Search engine from May 2010 to August 2010. We only used logs from registered users that provided birth date, gender and a valid zip code. We segment the log in the following age groups relying on the birth date:

- Young children: 8-9 years old
- Children: 10-12 years old
- Teenagers: 13-15 years old
- Adults: Above 18 years old

Similar age groups have been used in previous research on children search behavior on the web and represent marked stages of children development (Duarte Torres and Weber, 2011). We left out the group of users aged 16-18 for simplicity.

For each group of users we create a set of query tuples containing the query submitted by the user and a query reformulation which occurred within the same search session. In this work a search session is defined as a sequence of events $S = \langle e_1, e_2, .., e_n \rangle$ ordered in chronological order such that $timestamp(e_{i+1}) - timestamp(e_i) \leq 30$ minutes for every i. Each event can be either a query submission (e_i^q) or a click on a url (e_i^c). The time window length of 30 minutes is widely used in the literature (Huang and Efthimiadis, 2009; Jensen et al., 2006; Wang and Davison, 2008) and it has also been shown to be appropriate for search sessions of young users (Druin et al., 2009; Bilal, 2002).

A query $e_{i+1}^{q'}$ is a query reformulation of e_i^q if *(i)* the former is a prefix (e.g. *brit, britney spears*) or a suffix of the latter (e.g. *wars cheat codes, lego star wars cheat codes*), or the latter contains all the words of the former plus another word, independently of the order in which the words appears (e.g. *york giants, super bowl york giants xxv*) and *(ii)* there are no query events between them, although there can be arbitrary number of click events between the two queries (i.e. $S = \left\langle e_{i+1}^q, e_{i+2}^c, e_{i+3}^c, e_{i+4}^{q'} \right\rangle$ is allowed).

Using this procedure we obtained in the order of thousands of query tuples submitted by users aged 8 to 12 years old and hundreds of thousands for users above 12 years old.

We follow an analogous procedure to extract query tuples from the AOL search logs. Nonetheless since no user age information is provided in these logs we employed the methodology described by Torres et al. (2010) to extract search sessions with clicks landing on trusted content for children. These queries are identified by matching the urls clicked in the log with the domains listed in the Dmoz *kids and teens* section. Search sessions were grouped according the target audience of the urls (i.e. *kids*, *teens* and *adults*) instead of the user age. In Dmoz, urls tagged as for *kids* represent urls appropriate for children aged 8 to 12. With *teenagers* we refer to content appropriate for users aged 13 to 15 years old.

From the AOL logs were extracted around 480K queries and 20K sessions. From these sessions we obtained in the order of tens of thousands of query pairs for the three age groups.

6. Random Walk Evaluation

Assessing the quality of query suggestions can be a hard task given that the intent of the user is rarely clear from solely the query. However, we consider that the query suggestions that are frequently submitted by users of a given age range represent a good approximation of good query suggestions for this particular segment of users. A similar assumption has been adopted in previous query recommendation studies (Szpektor et al., 2011; Bing et al., 2011).

The performance of the query recommendation task was measured in terms of recall and NDCG. NDCG stands for Normalized Discounted Cumulative Gain, which is a measure of the effectiveness of IR systems in correctly ranking the results compared an optimal ranking. All the metrics are calculated based on the set of query tuples extracted and described in Section 5.2. Concretely, we have two datasets for testing: query tuples extracted from the AOL search logs, and query tuples extracted from the Yahoo! search logs.

To calculate the performance scores we define the set of query pairs from the gold standard as $G = \{\langle q, q' \rangle\}$ where q' is a query reformulation of q. And the set $S_n = \{\langle q, q', r \rangle\}$ where q' is a query suggestion of q and r is the ranked position of q'. For instance, recall is calculated as $r = \frac{|S_n \cap G|}{|G|}$. The intersection between the set of query suggestions and query reformulations was performed using exact matching.

Table 1 presents query examples of the query suggestions provided by our method, Bing and the gold standard for the queries extracted from the AOL and Yahoo! logs.

We compare the performance of the two variations of our method (Equation 3 and 4) for the forward and backward propagation schemes respectively) against the random walk baseline (Equation 2) established by Craswell and

`http://en.wikipedia.org/wiki/Discounted_cumulative_gain`

query	source	query suggestion
monsters	bing	truck games, jobs, high, jam, energy
	rw+kl	inc,music, film, pixar, images
	AOL	scary
sol practice	bing	in computer, fourth grade
quizzes	rw+kl	test, learning, history, science sites
	AOL	world history, history
free jigsaw	bing	online, online for adults, play, natgeo
puzzles	rw+kl	puzzles games, play , for kids, online
	Yahoo!	play
the cake	bing	t-shirt, meme, lyrics, song
is a lie	rw+kl	portal, guide, walk-through, games
	Yahoo!	portal

Table 1. Query examples from the query logs.

Szummer (2007) framework. Additionally, we compare the results obtained by our method against the Bing query suggestions. The evaluation was carried out using two test sets consisting of query pairs extracted from the AOL logs and from the Yahoo! search logs.

In a later stage of this work, we also compared the performance of our method with the three biased random walks described in Section 4 using the AOL search logs. This comparison was not carried using the Yahoo! search logs since this set was no longer accessible to any of the authors at this stage.

For all the methods two data models were employed: *kids* and *teens*. The model *kids* utilizes the domains from the Dmoz directory labeled as suitable for children up to 12 years old. The *teens* model employs the domains labeled as appropriate for users from 13 to 15 years old. Recall that the graph is constructed based on the seed urls from Dmoz.

The graph constructed with the *kids* model contains 91.6K edges and 20K nodes (12.9K urls and 7.1K tags). The graph teens contains 1.3M edges and 258K nodes (62.7K tags and 195.4K urls). In the tables shown in this section the baseline will be referred as *rw-b* and the two variations of our method as *rw-kl-f* and *rw-kl-b* respectively.

For each test set (AOL and Yahoo! search logs) we report two type of results: using pairs in which the reformulation land on a click and using pairs in which this click last at least 100 seconds, also referred as *long click* (Hassan et al., 2010). Long clicks have been shown to be a strong feature to predict search success. It has been widely reported (Craswell et al., 2008; Hua et al., 2011) that users tend to click on top ranked results even if these results do not contain the information that users are looking for. Duarte Torres and Weber (2011) showed that this behavior is even stronger for users of younger ages. The motivation behind using this restricted query set was to reduce this behavioral biased.

It is important to mention that there is a vocabulary gap between the training and test data since the datasets were extracted from different time windows. We found that the vocabulary intersection between the *children* query reformulations and the tag vocabulary was of 35% and 46% when using the *kids* and *teens* model. The intersection for *teenager* queries was slightly lower (32% and 41% respectively). Similar percentages were found in the AOL logs. These results indicate that the recall metrics obtained using these data sets are bounded to the percentages mentioned. All the results presented in the following sections were obtained by splitting the dataset in 10 folds and averaging the metric estimated (e.g. recall). The results reported were proven to be significant using the t-test with 0.01 level of confidence.

6.1. Experimental parameters

The parameters of our model and the biased random walks were set experimentally to maximize performance on a independent query sample extracted from the AOL log. The best settings were used to evaluate both query sets. For the case of our method we set the number of iterations of the random walk to *30* and we set the parameter α to *0.1*. The smoothing parameter μ of Equation 8 was set to *1200*. In the tables shown in this section the baseline will be referred as *rw-b* and the two variations of our method as *rw-kl-f* and *rw-kl-b*

query set	model	Bing	seedRank	TopicalRank	spamRank	rw-b	rw-k-f	rw-k-b	gain	gain-bias
				Top 5						
children	*kids*	1.5%	3.7%	2.3%	4.9%	3.3%	9.2%	9.5%	6.2%	4.6%
	teens		3.7%	2.3%	2.3%	2.4%	6.8%	7.4%	5.0%	5.1%
teenagers	*kids*	2.3%	3.7%	1.0%	2.8%	2.0%	5.2%	5.5%	3.5%	2.7%
	teens		4.3%	1.7%	3.9%	2.9%	8.0%	8.8%	5.9%	4.9%
adults	*kids*	5.1%	0.1%	0.1%	0.0%	0.3%	0.3%	0.4%	0.1%	0.4%
	teens		0.1%	0.1%	0.0%	0.4%	0.4%	0.4%	0.0%	0.4%
				Top 10						
children	*kids*	2.15	7.6%	2.7%	7.6%	4.1%	11.7%	12.1%	8.0%	4.5%
	teens		7.8%	3.5%	5.1%	4.5%	8.8%	9.7%	5.2%	4.6%
teenagers	*kids*	3.2%	6.3%	1.5%	3.9%	4.2%	6.5%	7.4%	3.2%	3.5%
	teens		7.2%	2.4%	4.2%	5.1%	10.5%	11.3%	6.2%	7.1%
adults	*kids*	5.1%	0.1%	0.1%	0.2%	0.7%	0.7%	0.7%	0.0%	0.5%
	teens		0.3%	0.2%	0.2%	0.9%	0.8%	0.8%	-0.1%	0.6%
				Top 50						
children	*kids*	2.1%	15.0%	9.6%	14.8%	13.2%	15.8%	16.6%	3.4%	1.8%
	teens		12.7%	8.2%	10.0%	6.9%	12.1%	12.4%	5.5%	2.4%
teenagers	*kids*	3.2%	8.1%	4.3%	8.1%	6.3%	9.9%	9.6%	3.3%	1.5%
	teens		9.6%	7.0%	11.3%	10.1%	12.7%	13.6%	3.5%	2.3%
adults	*kids*	5.1%	0.6%	0.5%	0.7%	1.0%	1.0%	1.0%	0.0%	0.3%
	teens		0.7%	0.8%	0.8%	1.3%	1.2%	1.3%	-0.1%	0.4%

Table 2. Recall comparison using the AOL log. Underlined values were not statistical significant when comparing our two random walks. The column *gain* expresses the performance difference between *rw-k-b* and *rw-b*. The column *gain-bias* refers to the difference between *rw-k-b* and *spamRank*.

query set	model	Bing	seedRank	TopicalRank	spamRank	rw-b	rw-k-f	rw-k-b	gain	gain-bias
				Top 5						
children	*kids*	0.017	0.028	0.025	0.037	0.021	0.069	0.071	0.050	0.034
	teens		0.027	0.031	0.040	0.018	0.042	0.045	0.027	0.005
teenagers	*kids*	0.024	0.029	0.013	0.030	0.016	0.042	0.048	0.032	0.018
	teens		0.035	0.019	0.021	0.031	0.064	0.067	0.036	0.046
				Top 10						
children	*kids*	0.026	0.048	0.032	0.053	0.032	0.082	0.086	0.054	0.033
	teens		0.049	0.039	0.051	0.021	0.053	0.055	0.034	0.004
teenagers	*kids*	0.031	0.042	0.016	0.036	0.017	0.045	0.046	0.029	0.010
	teens		0.049	0.022	0.029	0.036	0.075	0.078	0.042	0.049
				Top 50						
children	*kids*	0.026	0.081	0.052	0.076	0.076	0.089	0.091	0.015	0.015
	teens		0.069	0.055	0.068	0.042	0.069	0.071	0.029	0.003
teenagers	*kids*	0.031	0.049	0.023	0.047	0.029	0.045	0.050	0.021	0.003
	teens		0.059	0.034	0.049	0.069	0.076	0.081	0.012	0.032

Table 3. NDCG comparison using the AOL search logs.

respectively.

For the topical sensitive PageRank we set α to 0.3 and we employed 20 iterations. For the method proposed by Fuxman et al. (2008) we set α to 0.1 and we employed 25 iterations. For the method presented in Wu and Chellapilla (2007) we set the number of iterations to 25. The parameters found are similar to the ones reported by Haveliwala (2002), Fuxman et al. (2008) and Wu and Chellapilla (2007). We will refer to these methods in the AOL result tables as *topicalRank*, *seedRank* and *spamRank* respectively.

query set	model	Bing	seedRank	TopicalRank	spamRank	rw-b	rw-k-f	rw-k-b	gain	gain-bias
				Top 5						
children	*kids*	1.3%	3.6%	2.0%	3.3%	2.7%	8.9%	9.5%	6.7%	6.2%
	teens		3.3%	2.0%	5.3%	2.1%	6.6%	7.2%	5.1%	1.9%
teenagers	*kids*	2.3%	3.6%	1.2%	2.6%	1.5%	5.1%	5.5%	4.0%	2.9%
	teens		4.2%	1.9%	4.5%	2.4%	7.8%	8.8%	6.4%	4.3%
adults	*kids*	4.7%	0.0%	0.0%	0.0%	0.1%	0.3%	0.4%	0.3%	0.4%
	teens		0.0%	0.0%	0.0%	0.2%	0.3%	0.4%	0.3%	0.4%
				Top 10						
children	*kids*	1.8%	5.3%	2.6%	5.9%	3.6%	11.4%	12.0%	8.5%	6.1%
	teens		5.9%	2.6%	6.6%	4.2%	8.9%	9.5%	5.2%	2.9%
teenagers	*kids*	2.3%	5.6%	1.6%	3.5%	3.6%	6.4%	7.4%	3.8%	3.9%
	teens		6.5%	2.3%	5.8%	4.9%	10.4%	11.1%	6.3%	5.3%
adults	*kids*	4.7%	0.0%	0.0%	0.0%	0.2%	0.7%	0.7%	0.5%	0.7%
	teens		0.0%	0.3%	0.3%	0.5%	0.7%	0.7%	0.2%	0.4%
				Top 50						
children	*kids*	1.8%	13.2%	6.6%	10.5%	12.6%	15.6%	16.4%	3.8%	5.9%
	teens		11.8%	6.6%	13.2%	6.3%	12.1%	12.3%	6.0%	-0.9%
teenagers	*kids*	2.3%	7.8%	3.6%	7.8%	6.0%	9.8%	9.4%	3.4%	1.6%
	teens		9.0%	6.0%	12.4%	9.8%	12.5%	13.5%	3.7%	1.1%
adults	*kids*	4.7%	0.3%	0.0%	0.3%	0.6%	0.9%	1.0%	0.4%	0.7%
	teens		0.7%	0.7%	1.0%	0.7%	0.9%	1.3%	0.5%	0.3%

Table 4. Recall comparison using the AOL search logs (long clicks).

query set	model	Bing	seedRank	TopicalRank	spamRank	rw-b	rw-k-f	rw-k-b	gain	gain-bias
				Top 5						
children	*kids*	0.015	0.023	0.030	0.039	0.020	0.068	0.070	0.050	0.031
	teens		0.018	0.033	0.041	0.016	0.042	0.045	0.029	0.004
teenagers	*kids*	0.017	0.028	0.013	0.024	0.016	0.042	0.045	0.029	0.021
	teens		0.032	0.017	0.031	0.029	0.063	0.067	0.038	0.036
				Top 10						
children	*kids*	0.026	0.033	0.039	0.049	0.028	0.082	0.085	0.057	0.036
	teens		0.037	0.042	0.046	0.018	0.052	0.054	0.036	0.008
teenagers	*kids*	0.024	0.037	0.015	0.028	0.016	0.042	0.047	0.031	0.019
	teens		0.041	0.019	0.038	0.033	0.073	0.076	0.043	0.038
				Top 50						
children	*kids*	0.026	0.059	0.047	0.061	0.076	0.088	0.090	0.015	0.029
	teens		0.055	0.050	0.064	0.040	0.068	0.071	0.031	0.007
teenagers	*kids*	0.024	0.043	0.019	0.039	0.027	0.044	0.050	0.022	0.011
	teens		0.049	0.028	0.052	0.064	0.076	0.080	0.015	0.028

Table 5. NDCG comparison using the AOL search logs (long clicks).

6.2. AOL Query Log Results

Tables 2 shows the recall values obtained for the query pairs extracted from the AOL query log for the case of having landing clicks with the reformulation. We found that both variations of our method outperform the baseline and the Bing query suggestions for the *children* queries and the *teenager* queries. However, this is not the case for the set of *adults* queries, which was expected given that our random walk method give priority to tags that are more popular for children.

We observed that the maximum gain obtained is for the *children* queries when considering the top 10 results: 8.0% in respect to the baseline and 10.0% in respect to Bing. For the teenagers queries the maximum gain is also obtained at top 10 results using the *teens* model: 6.2% in respect to the baseline and 8.1% in respect to Bing.

We also found that for all the results the best performing model is aligned with the queries they target. For instance, the *kids* models perform better on queries targeting content for children and similarly for the *teens* model. This result is interesting because the kids model is considerably smaller than the *teens* models and yet

query set	model	Bing	rw-b	rw-k-f	rw-k-b	gain
			Top 5			
8-9	*kids*	3.6%	5.3%	11.1%	11.9%	6.6%
	teens		4.2%	8.0%	8.1%	3.9%
10-12	*kids*	3.7%	4.4%	8.3%	10.1%	5.7%
	teens		4.3%	5.4%	5.7%	1.4%
13-15	*kids*	4.1%	1.0%	3.0%	4.3%	3.3%
	teens		5.0%	8.6%	9.4%	4.4%
adults	*kids*	6.2%	0.3%	1.1%	2.1%	1.8%
	teens		0.4%	5.0%	5.8%	5.4%
			Top 10			
8-9	*kids*	7.6%	9.2%	14.0%	15.0%	5.8%
	teens		8.4%	12.2%	12.2%	3.8%
10-12	*kids*	7.6%	8.2%	12.1%	12.8%	4.6%
	teens		7.5%	8.0%	8.9%	1.4%
13-15	*kids*	7.9%	1.7%	7.0%	7.4%	5.7%
	teens		5.0%	10.2%	11.3%	6.3%
adults	*kids*	7.9%	3.2%	2.8%	3.5%	0.3%
	teens		6.5%	7.6%	7.8%	1.3%
			Top 50			
8-9	*kids*	7.6%	15.3%	21.0%	21.5%	6.2%
	teens		17.1%	22.1%	23.2%	6.1%
10-12	*kids*	7.6%	13.1%	16.3%	17.1%	4.0%
	teens		15.7%	20.6%	21.2%	5.5%
13-15	*kids*	7.9%	4.8%	13.1%	13.5%	8.7%
	teens		15.2%	17.4%	18.5%	3.3%
adults	*kids*	7.9%	4.3%	4.9%	5.6%	1.3%
	teens		7.1%	8.1%	8.9%	1.8%

Table 6. Recall comparison using the Yahoo! search logs.

query set	model	Bing	rw-b	rw-k-f	rw-k-b	gain
			Top 5			
8-9	*kids*	0.023	0.043	0.117	0.121	0.078
	teens		0.032	0.083	0.084	0.052
10-12	*kids*	0.039	0.044	0.105	0.110	0.066
	teens		0.037	0.052	0.053	0.016
13-15	*kids*	0.041	0.017	0.026	0.028	0.011
	teens		0.034	0.079	0.082	0.048
adults	*kids*	0.055	0.000	0.018	0.020	0.020
	teens		0.041	0.049	0.051	0.010
			Top 10			
8-9	*kids*	0.023	0.060	0.130	0.132	0.072
	teens		0.058	0.129	0.133	0.075
10-12	*kids*	0.040	0.061	0.119	0.121	0.060
	teens		0.051	0.066	0.069	0.018
13-15	*kids*	0.041	0.019	0.038	0.040	0.021
	teens		0.041	0.086	0.088	0.047
adults	*kids*	0.055	0.010	0.033	0.035	0.025
	teens		0.051	0.054	0.056	0.005
			Top 50			
8-9	*kids*	0.023	0.079	0.149	0.152	0.073
	teens		0.076	0.137	0.137	0.061
10-12	*kids*	0.040	0.079	0.135	0.137	0.058
	teens		0.080	0.106	0.109	0.029
13-15	*kids*	0.041	0.022	0.068	0.070	0.048
	teens		0.065	0.118	0.120	0.055
adults	*kids*	0.055	0.005	0.049	0.054	0.049
	teens		0.005	0.059	0.060	0.055

Table 7. NDCG comparison using the Yahoo! search logs.

query set	model	Bing	rw-b	rw-k-f	rw-k-b	gain
			Top 5			
8-9	*kids*	2.7%	4.7%	11.0%	11.7%	7.0%
	teens		4.2%	7.8%	7.9%	3.8%
10-12	*kids*	3.2%	4.1%	8.2%	8.8%	4.7%
	teens		4.1%	5.2%	5.6%	1.5%
13-15	*kids*	3.5%	0.8%	3.0%	4.3%	3.5%
	teens		4.7%	8.5%	9.3%	4.6%
adults	*kids*	5.7%	0.1%	0.9%	2.1%	2.0%
	teens		0.2%	4.8%	5.7%	5.6%
			Top 10			
8-9	*kids*	7.1%	8.7%	13.9%	15.0%	6.3%
	teens		8.3%	11.9%	12.0%	3.7%
10-12	*kids*	7.3%	7.6%	8.9%	9.9%	2.3%
	teens		7.0%	7.9%	8.9%	1.9%
13-15	*kids*	7.1%	1.3%	6.8%	7.3%	6.0%
	teens		5.0%	10.1%	11.2%	6.2%
adults	*kids*	7.2%	2.9%	2.6%	3.4%	0.5%
	teens		6.2%	7.3%	7.6%	1.5%
			Top 50			
8-9	*kids*	6.8%	14.9%	20.9%	21.5%	6.6%
	teens		17.1%	21.9%	23.2%	6.1%
10-12	*kids*	7.2%	12.7%	16.1%	17.1%	4.4%
	teens		15.4%	20.4%	21.1%	5.7%
13-15	*kids*	7.4%	4.6%	13.1%	13.5%	8.9%
	teens		14.9%	17.3%	18.3%	3.5%
adults	*kids*	7.2%	4.0%	4.8%	5.5%	1.5%
	teens		7.0%	8.0%	8.8%	1.8%

Table 8. Recall comparison using the Yahoo! search logs (long clicks).

query set	model	Bing	rw-b	rw-k-f	rw-k-b	gain
			Top 5			
8-9	kids	0.019	0.039	0.116	0.120	0.081
	teens		0.030	0.081	0.083	0.053
10-12	kids	0.033	0.042	0.105	0.109	0.067
	teens		0.035	0.049	0.051	0.017
13-15	kids	0.026	0.013	0.024	0.026	0.013
	teens		0.029	0.069	0.079	0.050
adults	kids	0.042	0.001	0.016	0.020	0.019
	teens		0.038	0.047	0.051	0.012
			Top 10			
8-9	kids	0.020	0.059	0.128	0.130	0.071
	teens		0.053	0.130	0.131	0.078
10-12	kids	0.034	0.057	0.115	0.120	0.063
	teens		0.051	0.067	0.067	0.016
13-15	kids	0.039	0.015	0.037	0.086	0.071
	teens		0.042	0.083	0.086	0.044
adults	kids	0.051	0.002	0.031	0.035	0.033
	teens		0.040	0.050	0.053	0.013
			Top 50			
8-9	kids	0.022	0.076	0.150	0.152	0.076
	teens		0.076	0.135	0.135	0.060
10-12	kids	0.038	0.077	0.134	0.138	0.061
	teens		0.077	0.105	0.108	0.031
13-15	kids	0.040	0.021	0.064	0.068	0.048
	teens		0.064	0.118	0.121	0.057
adults	kids	0.054	0.007	0.047	0.051	0.044
	teens		0.047	0.054	0.060	0.012

Table 9. NDCG comparison using the Yahoo! search logs (long clicks).

the recall scores are higher. This result suggests that simply adding web resources to the model may not lead to better results. Thus, the usefulness of the web resources seems to play an important role when ranking query suggestions.

From Table 2 we also observed that *rw-kl-b* performs consistently better than *rw-kl-f* and the performance difference in terms of recall between the two methods is at up 2.1%. The performance trends observed for the recall results were also reflected for the NDCG scores, as is shown in Table 3, which shows that the quality of the ranking is also improved by a reasonable margin.

It can be argued that these results are biased since the same collection of domains is utilized to extract the set of query pairs and to extract the set of bookmarks employed to construct the random walk graph (i.e. build the model). We show in the next section that all trends and results for the AOL logs also hold on the large set of query pairs extracted from Yahoo!.

We verified that the results reported in Table 2 and 3 were statistical significant by applying a paired t-test at the 0.01 level of confidence between the mean differences reported for all the possible pair of methods considered. We found that the differences reported between all the methods (e.g. *Bing* vs. *rw-b*, *Bing* vs. *rw-kl-f*, *rw-b* vs. *rw-kl-f*) were statistical significant for all the results reported. However, the exception was for the set of adult queries for which the difference between the methods *rw-kl-f* and *rw-kl-b* were not statistical significant. Values that were not proven statistical significant (between our two random walk variations) are underlined in the result tables.

Tables 4 and 5 show the results obtained for the set of AOL query reformulations in which the reformulation leads to a *long* click. All the methods obtained lower performance values. For instance for the *children* queries Bing obtains a recall of 1.8% at top 10 in contrast with the 2.1% obtained when using the first set of query pairs. Similarly the NDCG score obtained by Bing for the teenager set of queries (at top 10) is of 0.031 in the first set and 0.024 in the second. These results suggest that the problem of predicting query reformulation is harder when we are targeting reformulations that lead to long clicks. It is important to mention that even though we observed lower performance values, the ratio between our method and the two baselines were larger. For instance at top 10 the performance gain of rw-k-b in respect to *rw-b* was 8.5% and 10.2% in respect to Bing. Using the first set of query reformulations the performance gains were of 8.1% and 10.0% respectively. This result shows that our method performs better in the problem of predicting query suggestions that lead to long clicks which is convenient since these query suggestions have a higher likelihood of leading to relevant information.

6.2.1. Biased random walks results

In respect to the three biased random walks we observed that the *TopicalRank* has the lowest performance. We observed a performance loss of 1% to 3% in terms of recall in respect to *rw-b* (our first baseline). On the other hand we observed that the *seedRank* and *spamRank* perform similarly although the latter outperforms the former when considering the top 5 ranked query suggestions. These two methods outperform *rw-b* varying from 1% to 3% (in terms of recall) depending of the query set and the model employed. Nonetheless our method still performs better than all the biased random walks by a significant margin. In particular we observed that our method outperforms by a larger margin the biased random walks when considering the top 5 ranked results. For instance the recall gain with respect to *spamRank* for the *kids* model and children query set is of 4.6% and 4.4% at top 5 and top 10 respectively. The precision gain in respect to *rw-b* is of 6.2% and 8.0% respectively.

From the results obtained for the query suggestions landing on long clicks reported in Tables 4 and 5 we observed, on overall, lower performance for the three biased random walks considered, as it was the case for the Bing query suggestions and the *rw-b* method. This behavior was particularly noticeable when considering recall and NDCG at top 10. For instance for the children query set and for the *kids* model, the recall for *seedRank*, *TopicalRank* and *spamRank* was 3.7%, 2.3% and 4.9% in the query set with all the clicks while in the query set with long clicks was 3.6%, 2.0% and 3.3% respectively. At top 10, the performance varied from 7.6%, 2.7% and 7.6% to 5.3%, 2.6% and 6.6% for the three biased random walks respectively.

We found that the performance gain of our method in respect to the two best performing biased random walks for children queries tend to be higher when the evaluation is carried out on the query set with reformulations landing on *long* clicks. On the other hand we observed slightly lower performance gains for the teenager query set. For instance when using the best performing models (the model *kids* for the children query set and the

model *teens* for the teenager query set) the recall gain in respect to *spamRank* for the children query set is 6.2% at top 5 and 6.1% at top 10, while the recall gain observed for the query reformulations landing on clicks (not only long clicks) was of 4.6% and 4.5% a top 5 and top 10 respectively. For the case of the teenager query set, the gain varies from 4.3% to 4.9% for the query set on clicks and *long* clicks respectively when considering results at top 5, and 4.3% to 4.9% when considering results at top 10.

6.2.2. Yahoo! Search Engine Logs

Recall that the previous data set represent queries aimed at retrieving content for children and this dataset represent queries submitted by users for which their age can be estimated using the user profiles. Thus the results reported from this data provide a clearer picture of the performance of the methods for queries submitted by young users. Tables 6 and 7 report the recall and NDCG scores obtained by all the methods. As it was the case with the AOL query log, we observed that our random walk method outperforms the baseline and the query suggestions from Bing. Similarly *rw-kl-b* consistently performs better than *rw-kl-f*. The performance gain obtained by our method against the baseline and Bing was on the same order (around 6.1% with respect to the baseline and 9.2% with respect to Bing).

Interestingly, we found that the biggest performance gain is obtained for the youngest group of users and the gain decreases for older users. Another important difference observed with respect to the experiments on the AOL data is that the performance gain is higher at top 5 suggestions and not a top 10. This is a desirable result given the importance of ranking suggestions at the top positions in the case of children users (Duarte Torres and Weber, 2011). This result is also reflected by the low recall Bing has for the youngest group of users, particularly at top 5 (3.6% for users aged 8-9 vs 6.2% for adults). However, at top 10 the recall performance of Bing is comparable across all the age groups. This trend suggests that queries from the youngest are more frequently on the *long-tail* than queries from adults and teenagers. This observation emphasizes the usefulness of our method to address this type of queries since the best performing query suggestions for our method is obtained at top 5 for the youngest group of users.

As it was the case with the AOL data, the best performing model was aligned with its targeting age group. That is, the model *kids* which targets users under 12 years outperforms the *teens* model for queries of users from 8 to 12 years. Similarly, this model provided a better ranking quality, as it is shown in Table 7. We believe that this result is valuable because it suggests that the method proposed can be exploited on different information domains or on a different niche of users by carefully choosing the set of seed urls (i.e. model).

Tables 8 and 9 show the results obtained with the Yahoo! search logs using only the query reformulations that lead to *long* clicks. As we observed with the AOL logs, the performance values in terms of recall and NDCG were lower that the results obtained with the first set. Importantly, we also observed that the ratio gains between our method and the two baselines were also higher for these query reformulations, which suggests that the query suggestions provided by our method are more likely to lead to relevant content. For instance the performance gain in respect to Bing and *rw-b* for the queries of users from 10-12 years at top 10 was 7.9% and 6.3% respectively while the gain in the set of all clicks (not only *long* clicks) was of 7.4% and 5.8% respectively.

We apply the methodology described in the previous section to verify the statistical significance of the results. Similarly, we found all the results (when comparing all the combination of methods and within the models) were statistical significant at the 0.01 level. However, we found that for few cases the difference between our two random models were not statistical significant. These cases are shown underlined in Tables 6 and 7.

It is important to mention that the low values of NDCG reported are due to the sparsity of the data. On average we collected 1.6 query suggestions per query. Nonetheless, numbers on the same order have been reported on query recommendation studies for long-tailed queries (Szpektor et al., 2011).

7. Learning to Rank Tags

State of the art search engines employ large number of features to rank web results. In the previous section we showed that our method outperforms traditional random walks and a state-of-the art search engine in the problem of recommending queries to young users. We consider that the results can be improved further using a learning to rank framework in which our random walk method would represent one of the features. We envisage

three types of features to improve the system: language modeling, topic features, and distance to seed tags. In the following paragraphs we describe the features introduced for each one of these categories.

7.1. Language Modeling Features

We expect query suggestions to appear more frequently in the same documents in which the query occurs, particularly on documents written with a vocabulary appropriate for children. We build a language model using the websites listed in the Dmoz *kids and teens* directory to estimate the likelihood of a query suggestion candidate co-occur in the neighbourhood of the user's query. The language model is defined in the following way:

$$p(q|Mc_t) = \prod_{i=1}^{|Q|} p(q_i|Mc_t) \tag{20}$$

where Mc_t is the language model of the context in which the query suggestion t occur. The context is constructed using n-grams in a window of n words before and after q_i. Note that this window can be set as the whole document or as fine-grained section of the document. The probability is estimated in the following manner:

$$p(q_i|Mc_t) = \frac{cf_{Mc}(q_i, t) + \mu\ p(q_i|M)}{|C| + \mu} \tag{21}$$

where M is the entire Dmoz collection, $|C|$ is the total of n-grams in the context (i.e. pseudo document) we are considering and μ is the Dirichlet smoothing parameter. We estimate these parameters for optimal performance using a set of the queries of the AOL query log. The context window was set to 20 words before and after t and μ to 800.

We also expect suggestions more frequently used by children to appear more frequently in the Dmoz *kids and teens* collection. For this reason we consider as a feature the probability of the query suggestion in this collection: $p(t|M)$.

7.2. String features

We employed two simple string features: *(i)* query length and *(ii)* query suggestion length. We believe children favor shorter query suggestions since these suggestions tend to represent simpler words. We represent the length of the query by the number of tokens and by the number of alpha numerical characters. We report only the later since the results obtained by both approaches were identical.

7.3. Topic Features

We hypothesize that the rank of the suggestions can be improved informing the system about the topics that the query targets and the candidate suggestions that best represent the content of these topics. Consider the query *penguin*. The suggestions *games*, *online* and *cheats* are the 3 top ranked results provided by our random walk. Although, these suggestions are coherent and appropriate for children, the user submitting this query may be targeting general information about the flightless bird (e.g. for a school homework), or the user may simply be looking for pictures or videos of penguins instead of gaming related content. In the Dmoz kids & teens directory, penguin is associated to the topics *School Time\Science* and to *Games\Computers and Videos*. Using this information our system can boost the suggestions related to the topic *school time*, which are under-represented given the dominance of the gaming aspect for this query in our data.

The strategy is to generate a topic representation of the query and the candidate suggestion to estimate the topic coherence between the two elements. For this purpose, a query topic classifier was implemented by indexing the documents in the Dmoz *Kids and Teens* directory, which are appropriate for children up to 12 years old. We indexed around 15K websites with this approach. In this collection each document is located under one or more categories. Documents were mapped to topics by utilizing the most popular (in terms of size) and specific category to which the document belongs. We trim the depth of the category to two levels as an attempt to avoid data sparsity. This procedure led to 60 different categories or topics.

The classification of queries and query suggestions is carried out based on the top 100 matching documents (using standard *tf-idf* ranking). Concretely the topics associated to each one of the results returned for the query are

fetched and the retrieval score are aggregated on a per topic basis. In this manner we obtain a weight of the importance of each topic for the query. A vector of topic features is constructed by normalizing the scores in the vector between 0 and 1 (non-matching topics are assigned an score of zero). Formally, each topic feature is defined in the following manner:

$$cat_q(z) = \sum_{j=1}^{|R_{100}|} I(z = category(doc_j)) * score(doc_j) \tag{22}$$

where I is the identity function and $category(d)$ is the function that maps a document to a Dmoz category. The $cat_q(z)$ values are normalized by dividing over the sum of all the topic features calculated given a query q.

Using these features, each document is represented as the vector $V_Q = (cat_q(1), cat_q(2), .., cat_q(|z|))$ where $|z|$ is the total number of topics considered in the system.

Concretely we employed the topic vectors as features in two ways:

- the vector representation of the query as a set of features of size $|z|$
- by computing the cosine similarity between the topic vector of the query and the candidate query suggestion: $sim = \frac{\sum_{i=1}^{|z|} A_i \times B_i}{\sqrt{\sum_{i=1}^{|z|} A_i \sum_{i=1}^{|z|} B_i}}$

The motivation of the latter is to capture the topical cohesion between the query and the query suggestion.

7.4. Similarity to Seed Keywords

Seed tags as *for kids* and *kids* are often employed to signal that a web resource is designed for children. For this reason we expect children-oriented suggestions to appear more frequently near these seed tags. We estimate the distance between the candidate query suggestion and a pre-defined set of seeds tags (concretely, *for kids*, *for children*, *kids*, *4kids*). We expect this feature to stress candidate query suggestions that are more children friendly. The estimation was carried out by summing the probabilities of the query suggestion of being used to describe a web resource in conjunction with the seed tags:

$$sim(t_j, s) = \frac{\sum_{i=1}^{|s|} p(t_j|s_i)p(s_i)}{\sum_{j=1}^{|T|} \sum_{i=1}^{|s|} p(t_j|s_i)p(s_i)} \tag{23}$$

$$p(t_j|s_i) = \frac{cf(t_j, s_i)}{cf(s_i)} \tag{24}$$

where S is the set of seed tags. For each query the *sim* values of the query suggestion candidates are normalized between 0 and 1. The probability $p(t|s_i)$ is estimated on the delicious corpus which was also utilized for the random walk.

7.5. Learning to Rank Evaluation

We evaluate the features described in the previous section using the query set of users from 10-12 and 13-15 years old. Concretely, we employed a subset of the query reformulations utilized in the evaluation of the random walk. The training (and testing) data is in the order of the tens of thousands of query reformulations. The training data was constructed by extracting those queries for which there is at least one correct result in the list suggestions provided by the random walk, considering the top 50 ranked results. We use the gradient boosted regression tree learner to train the model and we perform 10-fold cross validation on the same data. Default parameters were used.

`http://people.cs.umass.edu/~vdang/ranklib.html`

feature	age	ndcg	ndcg-global
all	8-9	0.670	0.189
	10-12	0.642	0.172
	13-15	0.623	0.124
topic vector	8-9	0.137	0.230
	10-12	0.090	0.021
	13-15	0.111	0.022
topic sim	8-9	0.146	0.032
	10-12	0.123	0.028
	13-15	0.142	0.028
$p(q\|Mc_t)$	8-9	0.343	0.090
	10-12	0.313	0.070
	13-15	0.251	0.050
$p(t\|M)$	8-9	0.031	0.012
	10-12	0.052	0.014
	13-15	0.021	0.004
$sim(t\|S)$	8-9	0.052	0.016
	10-12	0.063	0.017
	13-15	0.051	0.015
q. length	8-9	0.006	0.002
	10-12	0.005	0.002
	13-15	0.003	0.001
s. length	8-9	0.001	0.000
	10-12	0.004	0.002
	13-15	0.003	0.001
rw-kl-b	8-9	0.564	0.132
	10-12	0.541	0.121
	13-15	0.523	0.082

Table 10. NDCG scores for each feature. Underlined values were not proven statistical significant at $p < 0.01$.

feature	age	ndcg	ndcg total	%diff
all	8-9	0.670	0.189	0.0%
	10-12	0.642	0.172	0.0%
	13-15	0.623	0.124	0.0%
no topic vector	8-9	0.642	0.181	-4.2%
	10-12	0.639	0.178	-0.5%
	13-15	0.602	0.143	-3.4%
no topic sim.	8-9	0.564	0.159	-15.8%
	10-12	0.550	0.153	-14.8%
	13-15	0.525	0.138	-15.7%
no $p(q\|Mc_t)$	8-9	0.555	0.162	-17.2%
	10-12	0.568	0.158	-11.5%
	13-15	0.581	0.138	-6.7%
no $p(t\|M)$	8-9	0.656	0.182	-2.1%
	10-12	0.612	0.162	-4.7%
	13-15	0.601	0.143	-3.5%
no $sim(t\|S)$	8-9	0.582	0.164	-13.1%
	10-12	0.572	0.159	-10.9%
	13-15	0.590	0.143	-5.3%
no q. length	8-9	0.670	0.189	0.0%
	10-12	0.641	0.171	-0.2%
	13-15	0.621	0.123	-0.3%
no s. length	8-9	0.670	0.189	0.0%
	10-12	0.640	0.171	-0.3%
	13-15	0.621	0.123	-0.3%
no rw-kl-b	8-9	0.172	0.069	-74.3%
	10-12	0.153	0.043	-76.2%
	13-15	0.160	0.038	-74.3%

Table 11. Leave-one-out NDCG scores. Underlined values were not proven statistical significant at $p < 0.01$.

7.5.1. Results

The best performance is obtained when all the features are combined: the NDCG is increased from 0.564 to 0.670, 0.541 to 0.642 and 0.523 to 0.623 for children aged 8 to 9, 10 to 12 and for teenagers respectively on the training data using 10-fold cross validation. The performance of the entire dataset (recall that the training data is a subset of the query reformulations extracted for users aged 10 to 12 years old) is increased from 0.132 to 0.189, 0.121 to 0.172 and 0.082 to 0.124 for users aged 8 to 9, 10 to 12 and 13-15 respectively.

Table 10 presents the NDCG scores obtained when each feature is employed independently. We found that the random walk score is by a large margin the most predictive feature (*e.g.* 0.541 vs. 0.313 of the next best performing feature in the case of users aged 10 to 12). The topic similarity metric and the language model trained on the Dmoz corpus were the next best performing features. For instance for users aged 10 to 12 these features represent a gain of 0.123 and 0.313 NDCG respectively. A similar behavior was observed for the other age groups.

The other features perform poorly when they are use in isolation in the system. This result can be explained by the fact that these features do not model the relation between the query and the query suggestion. For instance the feature expressed in equation 23 only provides information about the similarity of the query suggestion to a predefined set of seed tags. Nonetheless these features are beneficial when they are used in conjunction with the random walk score.

Table 11 presents the NDCG values obtained by the system when dropping each one of the features. Consistently with the results reported in Table 10, leaving out the random walk feature leads to a performance loss of 74.3% for children between 8 to 9 years old, 76.2% for users aged 10 to 12 and 74.3% for teenagers. Interestingly, the biggest performance loss after the random walk feature is obtained by dropping the topic similarity feature (*e.g.* 14.8% for the 10 to 12 group), which shows the importance of *informing* the system with the topical relation between the topics and the query. We observed that the topic representation of the query did not lead to significant improvements. This may be due to the large number of features that this vector represents (up to 80). Interestingly, the $sim(t|S)$ feature also leads to a significant loss of performance despite its simplicity (*e.g.* 13.1% for the 8 to 9 group). Given this results we believe it is worth to study in future work a more formal procedure to select the set of tags since the performance gain is already significant with a small (although representative) set of seed tags. As it was the case with the results in Table 10, we observed that the string features did not influence the overall performance of the classifier and consequently these features did not provided any performance gain in our experiments. The results presented in Table 10 and 11 were proven statistical significant using a t-test at $p < 0.01$ except for the values underlined in the tables.

8. Conclusions

In this paper we present how tags from social bookmarking system can be exploited to produce query suggestions for a specialized group of users using a set of seed web resources and a biased random walk based on the point-wise KL divergence metric between a foreground model and background model. We further improve the ranking of our results using a learning to rank approach to utilize the random walk score along intuitive features to boost query suggestions oriented on children topics. Our method can be used to improve current search assistance functionality for children since we show that our method performs the best for the youngest groups of users considered (8 to 9 years old). This segment of users is not served as well as older users, partly due to the long tail problem, which is more pronounced for these users according to the created test collection.

An important result of our work is the fact that we obtained consistent results in the AOL logs and the Yahoo! logs. This result suggests that a similar methodology to extract log data using quality seed web resources can be exploited to study and evaluate other problems in IR for children such as content filtering or re-ranking of children friendly web results. For future work we are interested in applying the method proposed in different domains. In this work we focused on content for children but potentially we could apply the same principles in domains such as query suggestions for professionals in the business domain or any other specific fields of expertise.

9. Acknowledgements

This research is funded by the European Community's Seventh Framework Programme FP7/2007-2013 under grant

References

Abou-Assaleh, T., Das, T., Gao, W., Miao, Y., O'Brien, P., and Zhen, Z.

Azzopardi, L., Dowie, D., Duarte, S., Eickhoff, C., Glassey, R., Gyllstrom, K., Hiemstra, D., de Jong, F., Kruisinga, F., Marshall, K., Moens, S., Polajnar, T., van der Sluis, F., and de Vries, A. (2012a). Emse: supporting children's information needs within a hospital environment. In *Proceedings of the 34th European conference on Advances in Information Retrieval*, ECIR'12, pages 578–580, Berlin, Heidelberg. Springer-Verlag.

Azzopardi, L., Dowie, D., and Marshall, K. A. (2012b). Yoosee: a video browsing application for young children. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '12, pages 1017–1017, New York, NY, USA. ACM.

Azzopardi, L., Dowie, D., Marshall, K. A., and Glassey, R. (2012c). Mase: create your own mash-up search interface. In *Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '12, pages 1008–1008, New York, NY, USA. ACM.

Baeza-Yates, R. and Tiberi, A. (2007). Extracting semantic relations from query logs. In *Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining*, KDD '07, pages 76–85, New York, NY, USA. ACM.

Bilal, D. (2002). Children's use of the yahooligans! web search engine iii. cognitive and physical behaviors on fully self-generated search tasks. *J. Am. Soc. Inf. Sci. Technol.*, 53(13):1170–1183.

Bing, L., Lam, W., and Wong, T.-L. (2011). Using query log and social tagging to refine queries based on latent topics. In *CIKM*, pages 583–592.

Boldi, P., Bonchi, F., Castillo, C., Donato, D., and Vigna, S. (2009). Query suggestions using query-flow graphs. In *Proceedings of the 2009 workshop on Web Search Click Data*, WSCD '09, pages 56–63, New York, NY, USA. ACM.

Craswell, N. and Szummer, M. (2007). Random walks on the click graph. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, SIGIR '07, pages 239–246, New York, NY, USA. ACM.

Craswell, N., Zoeter, O., Taylor, M., and Ramsey, B. (2008). An experimental comparison of click position-bias models. In *Proceedings of the international conference on Web search and web data mining*, WSDM '08, pages 87–94, New York, NY, USA. ACM.

Druin, A., Foss, E., Hatley, L., Golub, E., Guha, M. L., Fails, J., and Hutchinson, H. (2009). How children search the internet with keyword interfaces. In *IDC '09: Proceedings of the 8th International Conference on Interaction Design and Children*, pages 89–96, New York, NY, USA. ACM.

Duarte Torres, S., Hiemstra, D., Weber, I., and Serdyukov, P. (2012). Query recommendation for children. In *Proceedings of the 21st ACM international conference on Information and knowledge management*, CIKM '12, pages 2010–2014, New York, NY, USA. ACM.

Duarte Torres, S. and Weber, I. (2011). What and how children search on the web. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, CIKM '11, pages 393–402, New York, NY, USA. ACM.

Eickhoff, C., Serdyukov, P., and de Vries, A. P. (2010). Web page classification on child suitability. In *CIKM*, pages 1425–1428.

Feng, S., Lang, C., and Li, B. (2012). Towards relevance and saliency ranking of image tags. In *Proceedings of the 20th ACM international conference on Multimedia*, MM '12, pages 917–920, New York, NY, USA. ACM.

Fuxman, A., Tsaparas, P., Achan, K., and Agrawal, R. (2008). Using the wisdom of the crowds for keyword generation. In *Proceedings of the 17th international conference on World Wide Web*, WWW '08, pages 61–70, New York, NY, USA. ACM.

Gao, W., Niu, C., Nie, J.-Y., Zhou, M., Wong, K.-F., and Hon, H.-W. (2010). Exploiting query logs for cross-lingual query suggestions. *ACM Trans. Inf. Syst.*, 28:6:1–6:33.

Gyllstrom, K. and Moens, M.-F. (2010). Wisdom of the ages: toward delivering the children's web with the link-based agerank algorithm. In *Proceedings of the 19th ACM international conference on Information and knowledge management*, CIKM '10, pages 159–168, New York, NY, USA. ACM.

Gyllstrom, K. and Moens, M.-F. (2011). Clash of the typings: finding controversies and children's topics within queries. In *Proceedings of the 33rd European conference on Advances in information retrieval*, ECIR'11, pages 80–91, Berlin, Heidelberg. Springer-Verlag.

Gyöngyi, Z., Garcia-Molina, H., and Pedersen, J. (2004). Combating web spam with trustrank. In *Proceedings of the Thirtieth international conference on Very large data bases - Volume 30*, VLDB '04, pages 576–587. VLDB Endowment.

Hassan, A., Jones, R., and Klinkner, K. L. (2010). Beyond dcg: user behavior as a predictor of a successful search. In *Proceedings of the third ACM international conference on Web search and data mining*, WSDM '10, pages 221–230, New York, NY, USA. ACM.

Haveliwala, T. H. (2002). Topic-sensitive pagerank. In *Proceedings of the 11th international conference on World Wide Web*, WWW '02, pages 517–526, New York, NY, USA. ACM.

Haveliwala, T. H. (2003). Topic-sensitive pagerank: A context-sensitive ranking algorithm for web search. *Knowledge and Data Engineering, IEEE Transactions on*, 15(4):784–796.

Hua, G., Zhang, M., Liu, Y., Ma, S., and Ru, L. (2011). Automatically generating labels based on unified click model. In *Proceedings of the 20th international conference companion on World wide web*, WWW '11, pages 59–60, New York, NY, USA. ACM.

Huang, J. and Efthimiadis, E. N. (2009). Analyzing and evaluating query reformulation strategies in web search logs. In *CIKM '09*, pages 77–86, New York, NY, USA. ACM.

Hui, K., Gao, B., He, B., and Luo, T.-j. (2013). Sponsored search ad selection by keyword structure analysis. In *Proceedings of the 35th European conference on Advances in Information Retrieval*, ECIR'13, pages 230–241, Berlin, Heidelberg. Springer-Verlag.

Jensen, E. C., Beitzel, S. M., Chowdhury, A., and Frieder, O. (2006). Query phrase suggestion from topically tagged session logs. In *FQAS*, pages 185–196.

Kohlschütter, C., Chirita, P.-A., and Nejdl, W. (2007). Utility analysis for topically biased pagerank. In *Proceedings of the 16th international conference on World Wide Web*, WWW '07, pages 1211–1212, New York, NY, USA. ACM.

Li, M., Tang, J., Li, H., and Zhao, C. (2012). Tag ranking by propagating relevance over tag and image graphs. In *Proceedings of the 4th International Conference on Internet Multimedia Computing and Service*, ICIMCS '12, pages 153–156, New York, NY, USA. ACM.

Li, X., Snoek, C. G., and Worring, M. (2008). Learning tag relevance by neighbor voting for social image retrieval. In *Proceedings of the 1st ACM international conference on Multimedia information retrieval*, MIR '08, pages 180–187, New York, NY, USA. ACM.

Liu, D., Hua, X.-S., Yang, L., Wang, M., and Zhang, H.-J. (2009). Tag ranking. In *18th International World Wide Web Conference*, pages 351–351.

Ma, Y., Lin, H., and Lin, Y. (2011). Selecting related terms in query-logs using two-stage simrank. In *Proceedings of the 20th ACM international conference on Information and knowledge management*, CIKM '11, pages 1969–1972, New York, NY, USA. ACM.

Mei, Q., Zhou, D., and Church, K. (2008). Query suggestion using hitting time. In *Proceedings of the 17th ACM conference on Information and knowledge management*, CIKM '08, pages 469–478, New York, NY, USA. ACM.

Qiu, F. and Cho, J. (2006). Automatic identification of user interest for personalized search. In *Proceedings of the 15th international conference on World Wide Web*, WWW '06, pages 727–736, New York, NY, USA. ACM.

Ravi, S., Broder, A., Gabrilovich, E., Josifovski, V., Pandey, S., and Pang, B. (2010). Automatic generation of bid phrases for online advertising. In *Proceedings of the third ACM international conference on Web search and data mining*, WSDM '10, pages 341–350, New York, NY, USA. ACM.

Szpektor, I., Gionis, A., and Maarek, Y. (2011). Improving recommendation for long-tail queries via templates. In *Proceedings of the 20th international conference on World wide web*, WWW '11, pages 47–56, New York, NY, USA. ACM.

Torres, S. D., Hiemstra, D., and Serdyukov, P. (2010). Query log analysis in the context of information retrieval for children. In *SIGIR*, pages 847–848.

Van de Cruys, T. (2011). Two multivariate generalizations of pointwise mutual information. In *Proceedings of the Workshop on Distributional Semantics and Compositionality*, DiSCo '11, pages 16–20, Stroudsburg, PA, USA. Association for Computational Linguistics.

Wang, J. and Davison, B. D. (2008). Explorations in tag suggestion and query expansion. In *SSM '08: Proceeding of the 2008 ACM workshop on Search in social media*, pages 43–50, New York, NY, USA. ACM.

Wang, X. and Zhai, C. (2008). Mining term association patterns from search logs for effective query reformulation. In *CIKM '08*, pages 479–488, New York, NY, USA. ACM.

Wetzker, R., Zimmermann, C., and Bauckhage, C. (2008). Analyzing social bookmarking systems: A del.icio.us cookbook. In *Mining Social Data (MSoDa) Workshop Proceedings*, pages 26–30. ECAI 2008.

Wu, B. and Chellapilla, K. (2007). Extracting link spam using biased random walks from spam seed sets. In *Proceedings of the 3rd international workshop on Adversarial information retrieval on the web*, AIRWeb '07, pages 37–44, New York, NY, USA. ACM.

Yanbe, Y., Jatowt, A., Nakamura, S., and Tanaka, K. (2007). Can social bookmarking enhance search in the web? In *JCDL '07: Proceedings of the 7th ACM/IEEE-CS joint conference on Digital libraries*, pages 107–116, New York, NY, USA. ACM.

Zhang, X., Han, B., and Liang, W. (2009). Automatic seed set expansion for trust propagation based anti-spamming algorithms. In *Proceedings of the eleventh international workshop on Web information and data management*, WIDM '09, pages 31–38, New York, NY, USA. ACM.

Zhu, G., Yan, S., and Ma, Y. (2010). Image tag refinement towards low-rank, content-tag prior and error sparsity. In *Proceedings of the international conference on Multimedia*, MM '10, pages 461–470, New York, NY, USA. ACM.

Zhuang, J. and Hoi, S. C. (2011). A two-view learning approach for image tag ranking. In *Proceedings of the fourth ACM international conference on Web search and data mining*, WSDM '11, pages 625–634, New York, NY, USA. ACM.