A Collaborative Filtering Approach to Ad Recommendation using the Query-Ad Click Graph

Tasos Anastasakos, Dustin Hillard, Sanjay Kshetramade, Hema Raghavan
Yahoo! Inc
4402 Great America Parkway
Santa Clara, CA, 95054
{tasos,dhillard,sanjayk,raghavan}@yahoo-inc.com

ABSTRACT

Search engine logs contain a large amount of click-through data that can be leveraged as soft indicators of relevance. In this paper we address the sponsored search retrieval problem which is to find and rank relevant ads to a search query. We propose a new technique to determine the relevance of an ad document for a search query using click-through data. The method builds on a collaborative filtering approach to discover new ads related to a query using a click graph. It is implemented on a graph with several million edges and scales to larger sizes easily. The proposed method is compared to three different baselines that are state-of-the-art for a commercial search engine. Evaluations on editorial data indicate that the model discovers many new ads not retrieved by the baseline methods. The ads from the new approach are on average of better quality than the baselines.

Categories and Subject Descriptors

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

General Terms

Algorithms, Experimentation, Performance

Keywords

click-through rate, click data, collaborative filtering, online advertising, sponsored search

1. INTRODUCTION

Major search engines typically see traffic which amounts to several million queries and a correspondingly large number of user-clicks on documents as a response to those queries. While clicks are noisy due to issues of click fraud, accidental or exploratory clicks and position-bias, they can often be interpreted as a weak indicator of relevance. Several recent works have used click information to improve performance on various tasks (e.g., [11, 9, 18]). We propose an algorithm that operates on a large bipartite graph of queries and documents , where an edge between a query and a document is determined by whether users have clicked on the given document for the query. We apply an approach from collaborative filtering to first determine query-query similarity on this bipartite graph. The proposed method uses these query similarity scores to discover new documents that have not been shown for a given query before.

In this paper we illustrate the effectiveness of our proposed algorithm in the specific domain of sponsored search, where we aim to retrieve and rank textual ads that are relevant to a search query. Our work treats these textual ads as documents and in the remainder of the paper we use the terms ads and documents interchangeably. We find that our proposed method is better than several baseline systems in terms of the quality of the retrieved ads as measured by editorial assessments. Finally our proposed techniques are general enough so that they can be applied to other domains with click feedback such as web search or image search.

2. SPONSORED SEARCH OVERVIEW

A search engine typically displays sponsored listings on the top and the right hand side of the web-search results, in response to a user query. The advertiser bids on a set of keywords or **bidded phrases**, e.g., *sports shoes*, *canvas shoes* etc. The advertiser also submits a **creative** for each bidded phrase, which is a textual representation the search engine displays in response to a query. An advertiser can choose to use **standard** or **advanced** match for the keywords in an ad group. Enabling only standard match for the keyword "sports shoes", will result in the corresponding creative being shown only for that exact query. Whereas, if the keyword is enabled for advance match, the search engine can show the same ad for the related queries "running shoes" ,"track shoes" and other related queries. The algorithm proposed in this paper is used for "advanced match".

3. GENERATING AD SUGGESTIONS FROM THE QUERY-AD GRAPH

It is useful to describe the problem in terms of a bipartite graph $\mathcal{G}(\mathcal{Q}, \mathcal{A}, \mathcal{E})$. The set of queries $\mathcal{Q}, (|Q| = M)$ and the set of ads $\mathcal{A}, (|A| = N)$ comprise the partitions of the graph and $\mathcal{E}$ is the set of edges that connect queries to ads. A query q and an ad a are connected if a user issued the query q and clicked on the URL corresponding to the ad a from the list of sponsored search results. The edge between q and a is weighted by $r_{q,a}$ with $r_{q,a} > 0$. The weight $r_{q,a}$ represents the strength of the association and can be measured as a function of the number of clicks generated for query q and

CIKM'09, November 2–6, 2009, Hong Kong, China.
Copyright 2009 ACM 978-1-60558-512-3/09/11 ...$10.00.

ad a among all users, the click-through-rate (CTR) of the query ad pair, the number of conversions[1] for the query ad pair (q, a) or some other measure of affinity between query q and ad a. In this work an edge for (q, a) exists only if q and a appear in the sponsored search logs. Furthermore as described in Section 3, we use two formulations for the corresponding edge weight, a position normalized CTR (nCTR) and a machine-learned estimate of the probability of click.

The goal of the algorithm is to predict the unknown responses in the matrix R. Based on a common family of Collaborative Filtering approaches, termed "neighborhood methods" [4, 13, 17], our method first computes similarities between queries and then computes a prediction of the response between a query and a new ad based on how similar queries responded to the same ad. Breese et al [4] find that the correlation based formulation performs very well for a wide variety of tasks and our method is based on that. The approach consists of two operations: finding similar queries and then predicting the response of unseen query-ad pairs.

We estimate the similarity s_{ij} between two queries i and j with the Pearson correlation. If we represent queries by the corresponding rows in the response matrix R, and set non-observed query-ad pair responses to zero, the correlation similarity is computed as:

$$s_{i,j} = s(\boldsymbol{q}_i, \boldsymbol{q}_j) = \frac{\sum_{k\in\mathcal{S}(i,j)}(r_{i,k} - \overline{r}_i)(r_{j,k} - \overline{r}_j)}{\sqrt{\sum_{k\in\mathcal{S}(i,j)}(r_{i,k} - \overline{r}_i)^2}\sqrt{\sum_{k\in\mathcal{S}(i,j)}(r_{j,k} - \overline{r}_j)^2}} \tag{1}$$

where $\overline{r}_i$ ($\overline{r}_j$) is the mean response of query q_i (q_j) over all advertisements. The correlation measure is normalized for global mean and variance effects. The summations are computed over the set of common ads between the two queries, the support set $\mathcal{S}(i, j)$. Similarity measures computed over larger support sets are likely to be more robust, so we adjust the similarity metric by an overlap factor to discount similarity scores for query pairs with a small support set. The overlap factor is defined as:

$$J(\boldsymbol{q}_i, \boldsymbol{q}_j) = \frac{\sum_{k\in\mathcal{S}(i,j)}(r_{i,k} + r_{j,k})}{\sum_l r_{i,l} + \sum_l r_{j,l}}$$

The similarity score between two queries q_i and q_j becomes:

$$\widehat{s}(\boldsymbol{q}_i, \boldsymbol{q}_j) = J(\boldsymbol{q}_i, \boldsymbol{q}_j) \cdot s(\boldsymbol{q}_i, \boldsymbol{q}_j) \tag{2}$$

Each query is characterized by a vector $\boldsymbol{q}_i = [r_{i,1}, \ldots, r_{i,N}]$ where $r_{i,j}$ is the edge weight between query q_i and ad a_j. The semantics of the query q_i is captured in the ads that are associated with the query with varying measures of intensity represented by $r_{i,k}$ for each ad a_k. Similarly the queries that are associated with an ad in the bipartite graph provide a description of the ad. The similarity score sums over all ads in the support set. A potential disadvantage in this formulation is that ads that are linked to a very large number of queries have diffused semantics and are not a good indicator of similarity. We define the inverse frequency for advertisement a_k as: $f_k = \log \frac{M}{(\#\text{ of queries linked to } a_k)}$ and weigh the responses in $\boldsymbol{q}_i$ as: $\widehat{\boldsymbol{q}}_i = [(f_1 r_{i,1}), \cdots, (f_N r_{i,N})]$. The similarity score can now use the new edge weights $f_k r_{i,k}$ instead of $r_{i,k}$ in order to discount the effect of advertisers that associate the same ad with a large number of bidded phrases. These advertisers typically aim to display their ads to a large number of users to make their brand known and do not mind a low CTR, whereas our algorithm goal is to find relevant ads and queries. This approach is commonly used in information retrieval where word frequencies are modified by their inverse document frequency in order to offset the importance of words that occur in multiple documents.

Using the similarity measures defined above, we compute the set $Q^K(t)$ of K nearest neighbor queries to q_t. An ad a_i that is not adjacent to query q_t in the bipartite graph $\mathcal{G}$ has an unobserved response. If a_i is adjacent to any of the queries in $Q^K(t)$, we derive the predicted response $r_{t,i}$ between query q_t and ad a_i as the weighted average of the responses of these neighboring queries: $r_{t,i} = \frac{\sum_{k\in Q^K(t)} \widehat{s}_{t,k}\, r_{k,i}}{\sum_{k\in Q^K(t)} \widehat{s}_{t,k}}$. The observed responses $r_{k,i}$ are weighted by the query similarity score for each query rewrite q_t to q_k. If the response matrix R holds the observed CTR, the computed quantity is the predicted CTR for a new query-ad pair. For the remainder of the paper we refer to the response prediction following the above equation as the **QuAd-Click-Graph (query-ad-click-graph)** approach.

One issue with using observed CTR as a response meaure is the problem of *position-bias* for clicks [15, 11] which is partly due to the top-down order in which users navigate the ranked list and partly due to the inherent bias in their belief of a search engine's ability. To account for this position bias, we use a position-normalized CTR metric (nCTR) that computes the number of clicks divided by the expected clicks of an item as a rank normalized version of CTR [19, 7, 1].

The response $r_{k,i}$ can alternately be estimated using a machine learned model that predicts the probability that the user is likely to click on an ad for a query $P(click|q, ad)$. We find that learning a maximum entropy model for this task is quite effective. The model is identical to the one described in the work of Shaparenko et al[16] and is learned from the query logs of a major search engine. Empirically we found that using nCTR for the query-query similarity scores and $p(click|query, ad)$ for the query-ad similarity scores was the best weighting scheme. Detailed experiments on the choice of weighting schemes ($r_{k,i}$) for the query-query similarity method and query-ad similarity scores are available in [1].

The click-graph in our implementation is constructed from a portion of sponsored search traffic for a commercial search engine. The graph data is collected over a period of two weeks and consists of user queries and the associated ads that are displayed and clicked. A typical graph consists of 27 million unique queries, 20 million unique ads and 51 million query-ad edges. Each edge of this bipartite graph represents a click response between the associated query and ad. The graph is regenerated every week in order to include recent queries and capture recent changes in user activity. The algorithms can easily be implemented in a map-reduce framework and new query-query and query-ad associations can be detected in a few hours.

4. RELATED WORK

Past work on finding relevant ads for a query has typically used one of two different approaches: (a) query rewriting methods (e.g., [12, 14]) or (b) direct query-ad matching approaches (e.g., [5]). In query rewriting, the goal is to generate a relevant rewrite q_j for a given query q_i. Then ads associated with the bidded phrase q_j are retrieved in response to input query q_i. In direct query-ad matching the ads are treated as documents and are ranked using a

[1] number of clicks that ultimately resulted in a completed transaction, such as a purchase

standard information retrieval technique.

Typical query re-writing approaches learn from user query transformations extracted from web-search logs [12, 20]. These transformations include similar queries and sub-phrases in query reformulations that are obtained from user sessions in the logs. We use the session-based query rewrite approach [12] as a baseline and show that our proposed method performs well in comparison. It is also possible to use a traditional web-search approach (e.g., [6, 8]) for the advertising problem (e.g, [5]) . We further describe one such system that we use as our baseline in Section 5.

Recent work on query recommendation and query clustering has considered the input data in web search as a bipartite click graph of queries and documents with edges that correspond to click information (e.g., [3, 18, 9, 10, 2]). Viewed as a random walk on a bipartite graph, our method enables walks of 3-step transitions for query suggestions and 4-step transitions for query-ad response prediction. As found in prior work, limits on the maximum transition length lead to higher precision. Unlike most methods we utilize a function of CTR as the weight on the graph edges that avoids following popular paths obtained due to selection bias. A more detailed comparison of our work to random walk methods is deferred to the longer version of this paper [1].

Our work also follows prior research on collaborative filtering [4, 13, 17] that have been extensively applied to user-item data sets where the edge weights are either binary-valued or reveal ratings and preferences. The goals are to recommend new items to users preferences, cluster similar users or similar items based on associations derived from historical preferences. The use of Pearson correlation for recommender systems has been detailed in [4] and has been used in recommendation tasks for large data sets [13, 17].

5. EXPERIMENTAL SETUP

We compare the **QuAd-Click-Graph** method to three baselines. The first uses an information retrieval approach that matches queries directly to the ad and the remaining two baselines are based on query rewriting techniques.

Our first baseline is the direct query-ad system based on a commercial search index of the ads. Given a query, a search is performed on the index of landing pages and ad descriptions. The system uses a machine learned ranking function trained on query-document pairs similar to the the basic approach of Chen et al for web-search [8]. The model uses several features that broadly fall into three classes: (a) query features, such as query length (b) document level features, such as host-trust, anchor-text, category information and (c) features that model the query document relationship, such as word-overlap in various sections of the landing page. We refer to this as the **IRB (information retrieval baseline)** in the rest of the paper.

The first query rewriting baseline is the query log based system described in greater detail by the work of Jones et al [12] introduced in Section 4. We refer to this system as **LBQSB (log based query substitution baseline)**. The second query rewriting baseline is the collaborative filtering approach described in Section 3. We refer to this set of nearest neighbor queries as the **CFB (collaborative filtering baseline)**. We take up to five query rewrites per query, ranked by query similarity score.

The above three baseline methods are a subset of the techniques that contribute to edges on the click-graph used for our proposed methods. The Query-Ad Click Graph system builds on top of this graph and therefore, the method discovers not only new documents/ads not retrieved by the CFB method, but also new documents that were not discovered by any of the other baselines.

Method	% Relevant Listings	#listings	Average Score
IRB	68	311	0.39
CFB	54	908	**0.30**
LBQSB	45	683	**0.26**
QuAd-click-graph	67	1753	0.41

Table 1: Average relevance scores for offline editorial testing. Bolded numbers indicate the baselines which the QuAd-click-graph method outperformed in statistical significance tests.

We provide an evaluation of the **QuAd-Click-Graph** ad suggestions for a sample set of 1,000 randomly selected queries that are representative of typical search engine traffic. We generate the **QuAd-Click-Graph** ad suggestions as well as ads from each of the three baseline systems for this query set. The pooled set of query-ad pairs retrieved by all systems were labeled by trained editors who gave each ad a rating from one of the seven categories: the score *Perfect* is reserved for ads that are a perfect match to a query where there is only one correct answer. *Certainly Attractive* results meet or are strongly related to the likely commercial intent or explicit need of the query. *Probably attractive* and *Somewhat attractive* labels were assigned to ads that had a shift in scope/specificity from the original intent of the query. Judges marked listing as *Probably Not Attractive* when they can see the relationship between the listing and the query, but the listing is not likely to meet a commercial need of the user. Ads for which a user was never likely to click on were marked *Certainly Not Attractive*. The editors were experienced professionals, trained in the task, which had strict and very detailed guidelines with several examples for each label. Each judgment is associated with a point value (3.0, 1.0, 0.5, 0.4, 0.2, 0.0 respectively) which enables us to compute a weighted **average-score** of the listings returned by each of the systems.

6. RESULTS

Table 1 presents results of the editorial evaluation for query-ad pairs generated by our three baseline suggestion systems, as well as the proposed QuAd-click-graph. For each query, editors score all of the proposed suggestions and the scores are averaged to produce an overall quality score per method. Column 2 in table 1 gives the percent of retrieved ads that are relevant (ie., had a score greater than "somewhat attractive") and column 3 presents the total number of listings retrieved by each method.

The QuAd-click-graph query-ad pairs achieve the highest average editorial score, with a similar quality to the IRB system (although the IRB system produces much fewer overall ad suggestions). The CFB and LBQSB systems produce more suggestions (although still less than the QuAd-click-graph), but with significantly lower average quality.

The distribution of editorial scores per system is graphed in Figure 1. The bar graph displays the percent of suggestions that fall into each editorial category. Again, QuAd-click-graph and IRB have similar trends, with about 10% *Certainly Attractive* and 30% *Probably Attractive* ads. CFB

	$r_{t,i}$ threshold				
	0	.5	1	1.5	2
Total Listings	1,753	1,635	1,180	456	113
Total Queries	366	348	287	178	63
Average Score	0.41	0.41	0.45	0.51	0.62

Table 2: Average relevance scores for offline editorial testing of QuAd-Click-Graph with various thresholds on predicted response ($r_{t,i}$).

and LBQSB distributions are heavier at lower quality, with 20% to 30% receiving judgments of *Certainly Not Attractive* compared to about 10% for IRD and QuAd-click-graph. While the editors reviewed all ads with non-zero predicted response scores, we could filter what ads we present in live online tests. We investigate possible filtering thresholds on the predicted response scores ($r_{t,i}$) by computing the average editorial score for all ads that pass a given threshold. Table 2 presents the average editorial score at various response score thresholds. In addition, we present the number queries with ads (and total number of ads) at each threshold to show the trade-off between number of ads and average quality. We select a threshold of 1.5 in order to achieve an average editorial score of .51, while still maintaining coverage[2] of 178 out of 1,000 queries. Tests on online traffic revealed similar results and are reported in [1].

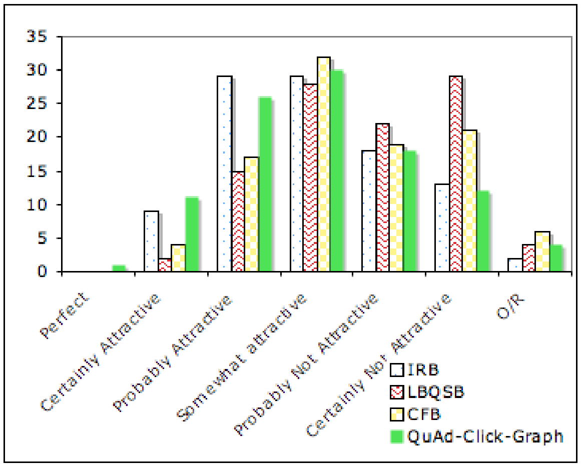

Figure 1: The fraction of judgments in each relevance category retrieved by each method.

7. DISCUSSION AND CONCLUSIONS

We have proposed a collaborative filtering algorithm for sponsored search ad retrieval that operates on a large bipartite graph of queries and ads to predict new ads likely to be clicked based on click data for related query-ad pairs. Our approach finds related queries based on a correlation measure over the query-ad graph, and then ranks candidate ads based on their average (weighted by query rewrite similarity score) expected click propensity over related queries. We compare our approach to three common baselines and find that we consistently improve performance. Editorial evaluations found good average quality for our method, and we verified the algorithm with online bucket tests. We find that ads proposed by our approach have a greater normalized click-through-rate and also perform well in the re-ranking stage of the ad presentation system.

8. REFERENCES

[1] T. Anastasakos, D. Hillard, S. Kshetramade, and H. Raghavan. A collaborative filtering approach to sponsored search. Technical Report YL-2009-006, Yahoo! Labs, 2009.

[2] I. Antonellis, H. G. Molina, and C. C. Chang. Simrank++: query rewriting through link analysis of the click graph. *Proc. VLDB Endow.*, 1(1), 2008.

[3] D. Beeferman and A. Berger. Agglomerative clustering of a search engine query log. In *KDD*, 2000.

[4] J. S. Breese, D. Heckerman, and C. Kadie. Empirical analysis of predictive algorithms for collaborative filtering. In *UAI*, 1998.

[5] A. Z. Broder, P. Ciccolo, M. Fontoura, E. Gabrilovich, V. Josifovski, and L. Riedel. Search advertising using web relevance feedback. In *CIKM*, 2008.

[6] C. Burges, T. Shaked, E. Renshaw, A. Lazier, M. Deeds, N. Hamilton, and G. Hullender. Learning to rank using gradient descent. In *ICML*, 2005.

[7] O. Chapelle and Y. Zhang. A dynamic bayesian network click model for web search ranking. In *WWW*, 2009.

[8] K. Chen, R. Lu, C. K. Wong, G. Sun, L. Heck, and B. Tseng. Trada: tree based ranking function adaptation. In *CIKM*, 2008.

[9] N. Craswell and M. Szummer. Random walks on the click graph. In *SIGIR*, 2007.

[10] A. Fuxman, P. Tsaparas, K. Achan, and R. Agrawal. Using the wisdom of the crowds for keyword generation. In *WWW*, 2008.

[11] T. Joachims, L. Granka, B. Pan, H. Hembrooke, and G. Gay. Accurately interpreting clickthrough data as implicit feedback. In *SIGIR*, 2005.

[12] R. Jones, B. Rey, O. Madani, and W. Greiner. Generating query substitutions. In *WWW*, 2006.

[13] Y. Koren. Factorization meets the neighborhood: a multifaceted collaborative filtering model. In *KDD-08*.

[14] F. Radlinski, A. Broder, P. Ciccolo, E. Gabrilovich, V. Josifovski, and L. Riedel. Optimizing relevance and revenue in ad search:a query substitution approach. In *SIGIR*, 2008.

[15] M. Richardson, E. Dominowska, and R. Ragno. Predicting clicks: estimating the click-through rate for new ads. In *WWW*, 2007.

[16] B. Shaparenko, O. Cetin, and R. Iyer. Data driven text features for sponsored search click prediction. In *AdKDD Workshop (KDD'09)*, 2009.

[17] A. Töscher, M. Jahrer, and R. Legenstein. Improved neighborhood-based algorithms for large-scale recommender systems. In *2nd Netflix-KDD Workshop (KDD'08)*, 2008.

[18] J.-R. Wen, J.-Y. Nie, and H.-J. Zhang. Clustering user queries of a search engine. In *WWW*, 2001.

[19] W. V. Zhang and R. Jones. Comparing click logs and editorial labels for training query rewriting. In *WWW 2007 Workshop on Query Log Analysis: Social And Technological Challenges*, 2007.

[20] Z. Zhang and O. Nasraoui. Mining search engine query logs for query recommendation. In *WWW*, 2006.

[2] coverage is the portion of queries for which ads from a given algorithm are available