Modeling Contextual Factors of Click Rates

Hila Becker
Columbia University
500 W. 120th Street
New York, NY 10027

Christopher Meek
Microsoft Research
1 Microsoft Way
Redmond, WA 98052

David Maxwell Chickering
Microsoft Live Labs
1 Microsoft Way
Redmond, WA 98052

Abstract

In this paper, we develop and evaluate several probabilistic models of user click-through behavior that are appropriate for modeling the click-through rate of items that are presented to the user in a list. Potential applications include modeling the click-through rates of search results from a search engine, items ranked by a recommendation system, and search advertisements returned by a search engine. Our models capture contextual factors related to the presentation as well as the underlying relevance or quality of the item. We focus on two types of contextual factors for a given item; the positional context of the item and the quality of the other results. We evaluate our models on a search advertising dataset from Microsoft's Live search engine and demonstrate that modeling contextual factors improves the accuracy of click-through models.

Introduction

The modern search engine uses ranking techniques in order to improve the quality and relevance of search results, in the way which would best reflect the user's intent. Researchers have used various different approaches to quantify this notion of relevance. The most obvious way to capture user preferences is by using human experts to explicitly evaluate search results. However, this approach is time consuming, and most users are reluctant to give this type of feedback due to the intrusive and prolonged nature of the task. On the other hand, the use of implicit feedback obtained from search engine logs is a more feasible technique for interpreting the user's link preferences. These logs are abundant as they are easy to collect, and they do not interfere with the user's search experience.

The problem with using query logs for making implicit inference of link relevance is that these logs capture an inherently biased view of user preferences. This bias may be due to several factors that are related to the original quality and presentation of search results from which the logs where generated, as well as the way in which the users examine the results. It was shown (Joachims et al. 2005) that the order of presentation of the search results has an effect on the click-through rates. Since users typically do not scan all of the search results before making their selection, it is important to consider the position of a link on a particular results page as a factor that may affect its click probability. It may also be the case that the probability that a link is clicked is affected by the quality of the alternative links in the result set. If the results above a particular link are relevant to the user's query, the probability that the link will be clicked may be affected by the fact that the previous search results already satisfy the user's search goals. We are interested in incorporating these types of external factors that bias the user selection when modeling click-through rates using query logs.

In this paper we look at click-through rates of search advertisements. Search advertisement query log data is especially useful for learning the effect of the link's position on click-through rates since the positions at which the advertisement links are shown exhibit greater variation than those of the main search results. If every link is always shown in a particular position, it is hard to separate the advertiser and positional effects on its click-through rate, unless we use some global measurement of positional effects. For these reasons, we focus on modeling click-through of search advertisement data. We perform our experiments on a search advertising dataset collected from logs of a major search engine.

In the first part of this paper, we consider two approaches for learning the click-through rate as a function of both the position and the advertisement itself. The first is a straight forward discriminative approach shown in a graphical form in figure 1a. The second approach, shown in figure 1b, assumes that there is a binary hidden variable which corresponds to the event that the user looks at a link. The idea behind the second approach is that users always look at a link before clicking, and whether or not they actually click on an advertisement link depends both on the link quality and the event that they looked at the link. The ”look” event in turn depends on the position at which the link is shown. We examine these alternative approaches against a baseline click-through rate calculation that does not incorporate any positional information.

In the second part of the paper we describe a new click-through rate prediction model, which incorporates contextual features such as relative placement in the result set and surrounding link quality. We compare our results across the different models that we considered in this paper.

Copyright © 2007, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

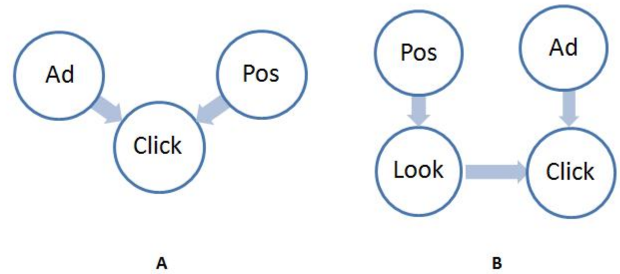

Figure 1: Graphical Representation of Click-though Models

Related Work

An important and active area of machine learning research focuses on leveraging user behavior as a source of implicit feedback to build and improve systems. For instance, Oard and Kim (1998) use implicit feedback to build recommender systems and Joachims (2002) uses click-through data to optimize a search engine. In these and other frameworks, users interact with a system through ranked lists of items. Examples of such ranked lists are search results from search engines, items from recommendation systems and advertisements from search-engine-advertisement systems.

A variety of implicit feedback techniques have been examined by different researchers. Oard and Kim (2001) present a general framework for modeling the content of information objects based on observation of user interaction with these objects when seeking information. They illustrate the potential of these techniques with respect to collaborative filtering, citation indexing and web search.

Fox et al. (2005) examine the correlation of different implicit measures of user search result preferences such as dwell time and scrolling events, to user satisfaction. Their study also showed a relationship between explicit and implicit measures of relevance and gave significant evidence that implicit measures are important in predicting user satisfaction, especially click-through data. White et al. (2005) also present results from a study which established that implicit relevance feedback is preferable to explicit relevance feedback, especially since users wish to reduce the burden of providing such feedback explicitly. Radlinski and Joachims (2005) use query chains, a sequence of reformulated queries, as input to an algorithm for inferring preference judgments. Agichtein et al. (2006) show improvement in click-through interpretation by modeling user behavior using different implicit measures as features. For a survey of other implicit feedback measures see Kelly and Teevan (2003) .

Our focus is on modeling click-through behavior of an item give its context. Click-through behavior is one of the most readily available and most powerful types of implicit feedback; clicks can be viewed as a passive endorsements of the items that are clicked. In addition, clicks can have intrinsic value to the system as is the case with search advertisements where advertisers are charged on the basis of user clicks on their advertisement (see Edelman et al. 2006).

Not surprisingly, the mode of presentation of the ranked lists of items affects the way in which users behave. In the context of search results, Joachims et al. (2005) , have demonstrated both a *positional* factor and a *contextual quality* factor in user behavior through their analysis of eyetracking data and click behavior. Agichtein et al. (2006) have shown similar effects for search results with click behavior and relevance data. In this paper, we develop and evaluate probabilistic models of user click-through behavior that are appropriate for modeling the click-through rate of items that are presented to the user in a list. We concentrate on developing models that incorporate both positional and contextual quality factors.

There is a large body of work on using click-through data as implicit feedback. Xue et al.(2004) use click-through data to improve search engine performance by viewing clicks as a way to associate a particular query with a set of links. We are interested in using click-through data for improving relevance of search advertisements.

Most of the work on using click-through data to improve search relevance has focused on the problem of extracting pairwise user preference from click-through logs. For example, Radlinski and Joachims (2006) introduce a method for generating bias-free training data by collecting pairwise link preferences. Agichtein et al. (2006) also evaluate their model on pairwise agreement between preferences and results.

Unlike this related work, our focus is on directly modeling the click-through of each item given its context. For some applications, one is interested in a quantitative measure of click-through rather than a qualitative preference relationship. In the case of search advertisements, for example, one needs models of click-through rate to compute expected revenue for alternative rankings. Furthermore, we suspect that in some ranking applications, it will be necessary to go beyond pairwise preferences to capture contextual quality factors and building click-through models to capture contextual effects is a first-step.

Notation

In this section, we describe the notation that we will use throughout the remainder of the paper. We use upper-case letters (e.g., X) to denote random variables. We sometimes use corresponding lower-case letters (e.g., x) to denote values of those variables.

We concentrate on the following search-application scenario: a user visits a search site and types in a particular query. In addition to the “organic” search results, a number of advertisement links are shown to the user. Our goal is to model the probability that the user will click on a particular advertisement link. We use the binary variable C to indicate whether or not the user clicks on the link, with $C = 1$ denoting a click, and $C = 0$ denoting a non-click. All of our models use, as input, the position of the link in the result set. We use the discrete variable P to indicate this position; the values of P distinguish among the possible placements of the link. For example, $P = p_1$ might indicate that the link was the top-most advertisement on the page. Finally, all of the models also use the advertiser identity of the link as input. We use the discrete variable A to denote the advertiser identity. The values of A correspond to the different advertisers whose advertisements can be shown to the user. For simplicity, we assume that any particular advertiser has exactly one advertisement that can be shown.

The probabilistic models that we describe in this paper define, for a given advertiser $A = a$ and a given position $P = l$, the probability that a user will click on that advertisement:

$$p(C = 1|A = a, P = l)$$

In Section 5, we will extend the models to include additional context about the advertisement link, such as the quality of the surrounding links and relative position on the results page.

Modeling the Location and Advertiser Effects

We consider two simple models for representing the advertiser and position effects on link click-through rates. The first model is a standard discriminative model that we learn using logistic regression. In our model, we have a separate feature for each advertiser and each position. In particular, we compute:

$$X(a, p) = \mu + \sum_{i=0}^{|A|} \alpha_i I(a, a_i) + \sum_{j=0}^{|P|} \lambda_j I(p, p_j) \quad (1)$$

where $I(x, y)$ is the indicator function that is one if $x = y$ and zero otherwise. Then, the click-through probability is modeled as

$$p(C = 1|A = a, P = l) = \frac{e^{X(a,p)}}{1 + e^{X(a,p)}}$$

To learn the parameters of the model (i.e., μ, $\alpha_1, ..., \alpha_{|A|}$, and $\lambda_1, \ldots, \lambda_{|P|}$), we apply a standard gradient descent algorithm; we use a Gaussian prior on the parameters that is centered on zero.

We can interpret the α parameters as the advertiser effects; if $\alpha_i > \alpha_j$, then for any particular position, the predicted click probability will be higher for advertiser a_i than for advertiser a_j. These α parameters may be potentially useful in detecting advertisement quality since they capture the effect of the advertiser alone on the click-through rate of the link. Similarly, we can interpret the λ parameters as positional effects. Experiments and results for this model are described in the following section.

For our second model, we introduce a hidden binary variable H and model the click probability as follows:

$$\begin{aligned} & p(C = 1|A = a, P = l) \\ & \quad = \quad p(C = 1|A = a, H = 1)p(H = 1|P = l) \quad (2) \end{aligned}$$

The second model is motivated by the intuition that before a user clicks on a link, he will almost certainly look at the advertisement first. If we interpret the event $H = 1$ as "the user looks at the advertisement", then our model can be understood as making the following assumptions: (1) the click event is independent of the position of the link once we know the advertiser of the link and we know that the user looked at the link, (2) whether or not a user looks at a link depends on the position of the link but not on the advertiser, and (3) the probability that the user clicks on the link without looking at the link is zero.

Using $\alpha_i = p(C = 1|A = a_i, H = 1)$ and $\lambda_j = p(H = 1|P = p_j)$, we see that the second model also has parameters corresponding to the advertiser effect and the location effect. We learn these parameters in the second model using a standard Expectation Maximization algorithm, with a uniform prior on the conditional probabilities (see Bilmes 1997).

The main difference between our two models is in the way that they treat the advertiser and location effects. The log-odds function $\log\left(\frac{p(C=1)}{p(C=0)}\right)$ used by the logistic regression model for prediction is a linear function of $X(a, p)$. In the hidden-variable model, on the other hand, the log probability $\log(p(C = 1))$ is linear in $X(a, p)$.

Experimental Comparison

In this section, we compare the two models described above using advertisement click-through log data from Microsoft's Live search engine (http://www.live.com). We train our models using data from the most frequent queries to our search engine and evaluate the performance of the models on our test data using log likelihood scores. We train various different "local" models using query-specific training data; each query has its own model with α and λ coefficients for the given query's advertisers and positions. We also train one "global" model using data across all queries, meaning that we use the same λ coefficients to predict the positional effect for all queries (similarly for α coefficients, although many advertisers are query-specific by nature). We use query-specific test datasets to test both the "global" model and the "local" model for the particular query. As a baseline experiment, we compute click-through rates by dividing the number clicks on an advertisement link by the number of impressions of that link.

We first look at the performance of the logistic regression

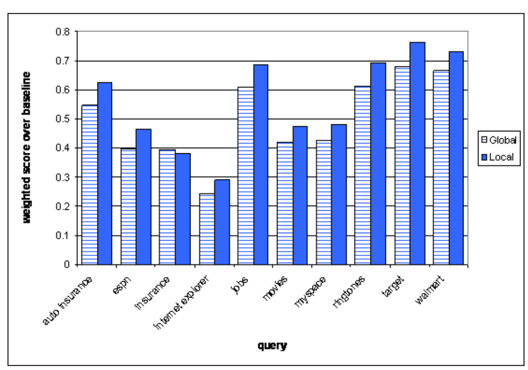

Figure 2: Local vs. Global Logistic Regression

models with respect to a baseline method, in terms of their log likelihood scores. Figure 2 shows the difference between the log likelihood of the logistic regression models and the baseline, weighted over the baseline, for a random subset of our datasets. Regardless of their performance with respect to the baseline, the local logistic regression models have better log scores than the global model across all of our test datasets. We will therefore focus our analysis on the local logistic regression models.

Figure 3 shows the difference between the log likelihood scores of the logistic regression models and a baseline click-through prediction method which captures no contextual effects. These log scores are computed for every one of our test datasets and are weighted over the baseline. All y-values greater than 0 imply that the logistic regression model outperforms the baseline. We can therefore see that our local logistic regression models are the best predictors of click-through rates among the approaches that we have considered so far. Using a sign test we determine that these results are statistically significant ($p < 0.0001$).

We also examine the performance of the hidden variable

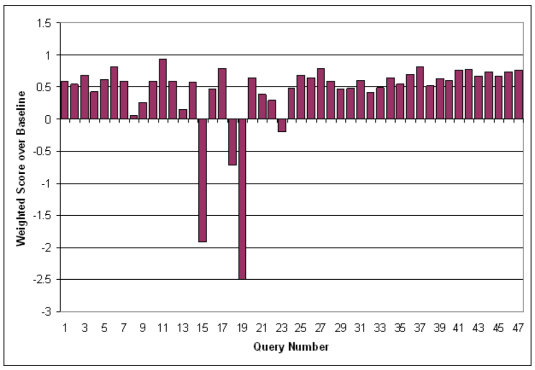

Figure 3: Logistic Regression Log Scores over the Baseline

model with respect to the baseline approach. By comparing the local and global hidden variable models we determine that once again the local models outperform the global models and therefore we focus on the local models exclusively. Figure 4 shows the difference between the log likelihood scores of the hidden variable model and the baseline, weighted over the baseline, for each one of our query-specific datasets. A positive difference implies better performance for the hidden variable model over the baseline for the given query. Using a sign test, the hidden variable model is also significantly better than the baseline with $p < 0.0001$. Next we compare the logistic regression and the hidden variable models against each other. Log likelihood scores of the logistic regression and hidden variable local models for a random subset of our datasets are presented in table 1. Using the log scores for all of our datasets, we test for statistical significance using a sign test and determine that the logistic regression models perform significantly better than the hidden-variable models with $p = 0.004$.

Since the local logistic regression models have higher log likelihood scores click-through rate predictors, we would like to examine them more carefully. The logistic regression coefficients can be useful in determining the contribution of each feature toward the event of a click. In order to analyze

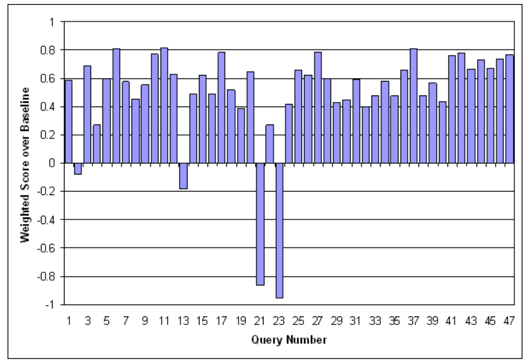

Figure 4: Hidden Variable Log Scores over the Baseline

Query	Logistic Regression	Hidden Variable
cars	-0.031022	-0.034205
chat	-0.058847	-0.060784
cingular	-0.101937	-0.103076
real estate	-0.052306	-0.060316
superman	-0.059982	-0.085866
weather	-0.089299	-0.090457

Table 1: Logistic Regression and Hidden Variable Log Scores

the positional effect on click-through, we look at the logistic regression coefficients for the different positions. Table 2 contains the positional coefficients of the logistic regression models for several queries. A larger value for the coefficient implies a greater contribution to the event $C = 1$. Note that the general trend shows that higher positions have larger coefficient values, implying that a user is more likely to click on links that are displayed at higher positions in the result set. Values that do not follow the trend may be attributed to particularly relevant links appearing frequently in the same position. Another possible factor may be related to the relative position of the advertisement link on the page.

Query	pos 1	pos 2	pos 3	pos 4
amazon	-1.362	-2.017	-1.231	-3.919
debt consolidation	-0.921	-1.312	-1.365	-1.850
games	-3.434	-3.805	-3.679	-5.415
mp3	-0.507	-1.062	-1.580	-2.569
ringtones	-1.198	-1.226	-1.481	-2.954
white pages	-1.161	-0.407	-1.194	-4.901

Table 2: Logistic Regression Position Coefficients

Given that the logistic regression results proved to be significantly better than the baseline approach and the hidden-variable model, we extend this logistic regression model in the next section, to include contextual features such as surrounding link quality and relative placement.

Adding Contextual Features

So far we only considered the effect of the advertiser and the position on click-through rate prediction. In this section, we expand our feature set to include additional contextual information regarding the advertisement's relative position and quality. The new features are meant to capture the quality of the alternative links; intuitively, if the link of interest is placed below a particularly good advertisement link, we expect that the click rate will be lower than if it is placed below a particularly bad advertisement link. Given the results in the previous section, we consider the added contextual features only in the logistic-regression model; for future work, we plan to incorporate these features into the hidden-variable model as well.

In our search engine, advertisement links are displayed either on the main line of the results page, preceding the organic results, or on a side bar located on the right hand side of the page. The number of advertisements shown on the main line and the side bar varies and sometimes there are no main line or side bar links at all. In the new model, we use this distinction between the two groups of links since their different locations on the results page may affect click-through. An example of the advertisement links layout on a results page can be seen in figure 5.

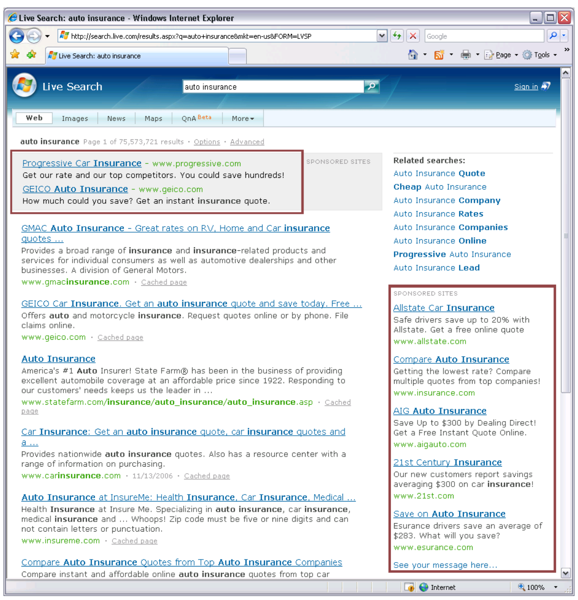

Figure 5: Search Results with Advertisement Links

The new features that we add to our models include contextual information denoting relative placement of advertisement links in the result set, as well as different functions of the advertiser-effect parameters of alternative links from the logistic regression model. In particular, we consider including features such as the link's physical position on the page: main line versus side bar, whether the quality of link above/below is better and the number of links above/below which are better.

We extend the logistic regression model to include these features as follows:

$$X(a,p) = \mu + \sum_{i=0}^{|A|} \alpha_i I(a, a_i) + \sum_{j=0}^{|P|} \lambda_j I(p, p_j) + \sum_{k=0}^{|Q|} \theta_k Q_k(a,p) \tag{3}$$

A graphical representation of this model can be seen in figure 6.

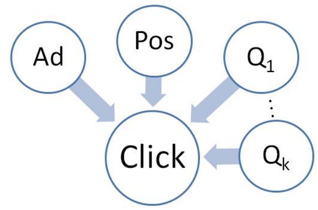

Figure 6: Contextual Model - Graphical Representation

Learning the model parameters of the extended logistic-regression model is problematic because the quality features are functions of the α parameters; we can no longer apply standard methods to learn the parameters. We apply the following boosting-style algorithm to learn the parameters of the extended model. First, we set $Q_k = 0$ for all k, and run the standard gradient-descent algorithm. Next, we update the Q_k values using the α values that resulted. We then re-run the gradient-descent algorithm to learn new values for the parameters, keeping the Q_k values constant. We iterate this process of defining constant Q_k values in terms of the α values from the previous iteration. After each iteration, we record the log-likelihood of the data. We stop this process after the first iteration in which the log-likelihood decreases, and we use as our final model the one from the previous iteration in which the log-likelihood was the highest.

We compare the results of the simple local logistic regression model, our best performing model so far, to the contextual model that was trained using the gradient descent algorithm. From table 3 we observe that while there is a difference in performance between the two approaches, it is not as apparent as it was with the previous improvements to the baseline approach.

Query	**Logistic Regression**	**Contextual Model**
astrology	-0.012498	-0.012285
britney spears	-0.049658	-0.050804
debt	-0.013555	-0.012102
expedia	-0.152005	-0.151790
insurance	-0.023235	-0.018511
music	-0.010922	-0.010875

Table 3: Logistic Regression and Contextual Model Log Scores

Figure 7 shows the weighted relative gain of the contextual model with respect to the simple local logistic regression model per query. While the contextual models perform significantly better than the simple logistic regression using a sign test ($p < 0.04$), they do not yield a very large gain with an average of ≈ 0.01 weighted difference. This implies that from a practical standpoint, it may be beneficial to forgo the extra contextual features in favor of the simpler model in order to improve complexity.

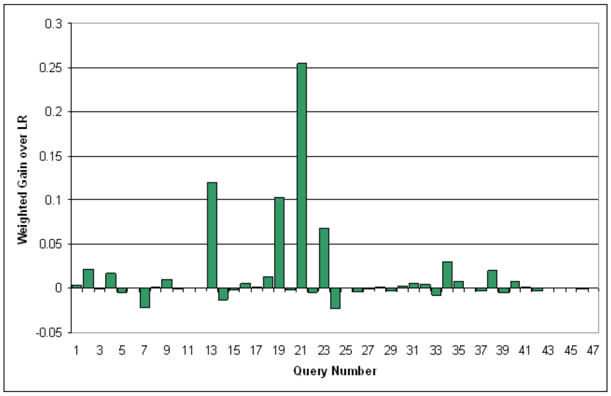

Figure 7: Contextual Model Gain over Logistic Regression

Conclusion

In this paper we presented several approaches for modeling the click-through rates of advertisement links on a large-scale search engine. In such an application, it is particularly important to learn parsimonious models that capture the presentation and advertiser effects on click-through rates; the choices that we make for which advertisements to show in what contexts have an immediate impact on the amount of revenue that we gain from the system.

We considered two simple models for learning the advertiser and positional effects, and compared their performance to a standard method for click-through rate prediction. Both of our modeling approaches, logistic regression and hidden "look" variable, proved to be significantly better than the baseline with $p < 0.001$ using a sign test. Since the success of both approaches shows that modeling the advertiser and position information improves click-through rate prediction, we were interested in extending one of these models to include other contextual information. After evaluating the two approaches against each other, we determined that the logistic regression models were the better click-through rate predictors.

Using this information, we decided to extend the logistic regression models to include contextual features such as relative position on the page and quality of the alternative links in the result set. Although the results are statistically significant, the relative difference between the log likelihood scores is not a large one. We note here that in practical applications one should consider the tradeoff between the accuracy and complexity of these approaches. For instance, given the small gain from the additional contextual features and scalability issues for using local models, a global logistic regression model trained using a sample of the data may be the best approach.

One limitation of the contextual model is that it can only learn about differences that appear in data. For future work, we plan to study the trends in our data as well as expand our datasets in order to identify other useful features to model. We would also like extend the hidden-variable model to include contextual features, and determine if a different modeling approach could yield a significant performance gain. Other directions for future work include utilizing click information about surrounding advertisement links in training a predictive model of click-through, developing models for predicting the probability of the set of click-through sessions for a page, and modeling the probability of click-through using features of the users previous behavior.

References

Agichtein, E.; Brill, E.; Dumais, S. T.; and Ragno, R. 2006. Learning user interaction models for predicting web search result preferences. In *SIGIR*, 3–10.

Agichtein, E.; Brill, E.; and Dumais, S. T. 2006. Improving web search ranking by incorporating user behavior information. In *SIGIR*, 19–26.

Bilmes, J. 1997. A gentle tutorial on the em algorithm and its application to parameter estimation for gaussian mixture and hidden markov models.

Chickering, D. M. 2002. The winmine toolkit.

Edelman, B.; Ostrovsky, M.; and Schwarz, M. 2006. Internet advertising and the generalized second price auction: Selling billions of dollars worth of keywords.

Fox, S.; Karnawat, K.; Mydland, M.; Dumais, S. T.; and White, T. 2005. Evaluating implicit measures to improve web search. *ACM Trans. Inf. Syst.* 23(2):147–168.

Joachims, T.; Granka, L. A.; Pan, B.; Hembrooke, H.; and Gay, G. 2005. Accurately interpreting clickthrough data as implicit feedback. In *SIGIR*, 154–161.

Joachims, T. 2002. Optimizing search engines using clickthrough data. In *KDD*, 133–142.

Kelly, D., and Teevan, J. 2003. Implicit feedback for inferring user preference: a bibliography. *SIGIR Forum* 37(2):18–28.

Oard, D. W., and Kim, J. 1998. Implicit feedback for recommender systems.

Oard, D. W., and Kim, J. 2001. Modeling information content using observable behavior.

Radlinski, F., and Joachims, T. 2005. Query chains: Learning to rank from implicit feedback. In *KDD*, 239–248.

Radlinski, F., and Joachims, T. 2006. Minimally invasive randomization fro collecting unbiased preferences from clickthrough logs. In *AAAI*.

White, R. W.; Ruthven, I.; and Jose, J. M. 2005. A study of factors affecting the utility of implicit relevance feedback. In *SIGIR*, 35–42.

Xue, G.; Zeng, H.; Chen, Z.; Yu, Y.; Ma, W.; Xi, W.; and Fan, W. 2004. Optimizing web search using web clickthrough data. In *CIKM*, 118–126.