Improving Ad Relevance in Sponsored Search

Dustin Hillard, Stefan Schroedl, Eren Manavoglu,
Hema Raghavan and Chris Leggetter
Yahoo! Labs
Great America Parkway
Santa Clara, CA, 95054
{dhillard,stefan,erenm,raghavan,cjl}@yahoo-inc.com

ABSTRACT

We describe a machine learning approach for predicting sponsored search ad relevance. Our baseline model incorporates basic features of text overlap and we then extend the model to learn from past user clicks on advertisements. We present a novel approach using translation models to learn user click propensity from sparse click logs.

Our relevance predictions are then applied to multiple sponsored search applications in both offline editorial evaluations and live online user tests. The predicted relevance score is used to improve the quality of the search page in three areas: filtering low quality ads, more accurate ranking for ads, and optimized page placement of ads to reduce prominent placement of low relevance ads. We show significant gains across all three tasks.

Categories and Subject Descriptors

H.3.3 [**Information Retrieval**]: Information filtering; I.5.4 [**Pattern Recognition**]: Applications—*Text processing*

General Terms

Algorithms, Experimentation

Keywords

advertising, relevance modeling, clicks, translation

1. INTRODUCTION

Large commercial search engines typically provide organic web results in response to user queries and then supplement with sponsored results that provide revenue based on a "cost-per-click" billing model. Sponsored results are selected from a database populated by advertisers that bid to have their ads shown on the search result page. A search engine typically decides which ads to show (and in what order) by optimizing revenue based on the probability that an ad will be clicked, combined with the cost of the ad [29]. Beyond selecting and ranking potential ads, a search engine also must decide how many ads to show and how prominently (such as above the search results, or at the side). A search engine could likely increase short term revenue by increasing the number and prominence of sponsored results, but such an approach typically would reduce overall quality and eventually result in users switching to another search engine. Each search engine chooses how aggressively to advertise based on a balance of business goals that incorporate revenue and estimated user impact.

While adding the perfect advertisement to a search result page may actually improve user experience, most search engine users find that the quality of sponsored links somewhat degrade the search experience on average. Previous work in sponsored search has primarily described modeling clicks [29, 11], but in this work we focus on predicting ad relevance in order to automatically identify low relevance ads. The approach more closely resembles the typical information retrieval ranking task, which aims at predicting document relevance (rather than directly modeling the probability that a user will click on a document). Given our predicted relevance we then proceed to alter multiple aspects of the sponsored search system with the goal of improving overall quality. We measure the improvement offline with editorial analysis and online with live tests over millions of users.

We specifically model ad relevance in order to facilitate improving the sponsored search system. While relevance and clicks are highly related there are important differences. Editorial assessment of relevance typically captures how *related* an advertisement is to a search query, while click-through-rate (CTR) provides a signal about if an ad is *attractive*. The two measures can diverge: an ad to "Buy Coke Online" is highly related to the search "cocacola" although the CTR is low because very few people are interested in buying Coke over the Internet, conversely an ad for "Coca Cola Company Job" is less related to the query but obtains a much higher CTR in our logs because the ad is highly attractive to users. A more drastic example is an ad to "Lose weight now" that receives a large number of clicks independent of what search term the ad is shown for (in most cases the ad would be judged to have low relevance to any particular search term). Previous sponsored search work primarily models click probabilities in order to estimate expected revenue and rank candidate ads. In this work we concentrate on additionally predicting ad relevance in order to improve our ability assess and optimize the sponsored search system.

In Section 2 we review the sponsored search problem and in Section 3 we describe our baseline relevance model. Sec-

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
WSDM'10, February 4–6, 2010, New York City, New York, USA.
Copyright 2010 ACM 978-1-60558-889-6/10/02 ...$10.00.

tion 4 describes experiments that incorporate user clicks as features to improve relevance modeling. Section 5 presents results leveraging the predicted relevance score for three sponsored search applications. Section 6 discusses related work and Section 7 summarizes our findings.

2. REVIEW OF SPONSORED SEARCH

Search engines typically display sponsored listings on the top (north) and the right hand side (east) of the web-search results, in response to a user query. The revenue model for these listings is "cost-per-click" where the advertiser pays only if the advertisement is clicked. The advertiser "targets" specific keyword markets by bidding on search queries. For example, an advertiser selling shoes may bid on user queries such as "cheap shoes", "running shoes" and so on. Sponsored search offers a more targeted and less expensive way of marketing for most advertisers as compared to media like TV and newspapers and has therefore gained momentum in the recent few years, becoming a multi-billion dollar industry.

We now describe the search engine monetization (SEM) terminology used in this paper. An advertising campaign consists of many *ad groups* where each ad group in turn consists of a set of *bidded phrases* or keywords that the advertiser bids on, e.g., "sports shoes", "stilettos", "canvas shoes", etc. A *creative* is associated with an ad group and is composed of a *title*, a *description* and a *display URL*. The title is typically 2-3 words in length and the description has about 10-15 words. Clicking on an ad leads the user to the *landing page* of the advertiser. An advertiser can choose to use *standard* or *advanced* match for the keywords in an ad group. For example, enabling only standard match for the keyword "sports shoes" will result in the corresponding creative being shown only for that exact query. If the keyword is enabled for advanced match, the search engine can show the same ad for the related queries "running shoes" or "track shoes." A *bid* is associated with each keyword and a second price auction model determines how much the advertiser pays the search engine for the click [12].

Most search engines typically take a three-stage approach to the sponsored search problem: (1) finding relevant ads for a query, (2) estimating click through rate (CTR) for the retrieved ads and appropriately ranking those ads, and (3) selecting how to display the ads on the search page (i.e. how many ads to show in the north section).

Finding relevant ads to a query is an information retrieval problem and the nature of the queries makes the problem quite similar to web search. Yet, there are some key differences between web search and sponsored search. One of the primary differences is that the collection of web documents is significantly larger than the advertiser database, and retrieving candidate ads for infrequent queries is a very important area of research for sponsored search. In addition, sponsored results may relate to the search in a more broad sense than would be reasonable for web results. For example an ad for "Limo Rentals" would be relevant to a search for "prom dress" although it would not likely be a reasonable top organic web result.

After retrieving a set of ads $\{a_1 ... a_n\}$ for a query q shown at ranks $1...n$ on search results page, the expected revenue is given as:

$$R = \sum_{i}^{n} P(click|q, a_i) \times cost(q', a_i, i) \quad (1)$$

where $cost(q', a_i, i)$ is the cost of a click for the ad a_i at position i for the bidded phrase q'. In the case of standard match $q = q'$. Most search engines rank the ads by the product of the estimated CTR, $P(click|q, a_i)$, and bid in an attempt to maximize revenue for the search engine. Therefore, accurately estimating the CTR for a query-ad pair is a very important task that has significant revenue implications. One simple approach is to use the observed historical CTR statistics for query ad pairs that have been previously shown to users. However, the ad inventory is continuously changing with advertisers adding, replacing and editing ads. Likewise, many queries and ads have few or zero past occurrences in the logs. These factors make the CTR estimation of rare and new queries a challenging problem.

When a set of ads has been retrieved and ranked the search engine must then decide how many ads to show, and where to place the ads on the search result page. Many queries do not have commercial intent, so displaying ads on the top of a page for a query like "formula for mutual information" may hurt user experience and occupy real-estate on the search results page in a spot where a more relevant web-search result might exist. Therefore, in sponsored search, we prefer not to show any ads when the estimate of CTR and/or relevance of the ad is low. Using the same user experience argument, for a navigational query [3] like "bestbuy.com", we would rather show only that particular retailer's website if that ad existed in the advertiser database. We refer the reader to the study of Jansen and Resnick [14] for further details on user perceptions of sponsored search. Determining how many candidates to retrieve and display is less crucial in web search because the generally accepted user model is one where users read the page in sequence and exit the search session when their information need is satisfied. In sponsored search the search engine must decide how many ads to place in the north page section above the web results as well as the total number of ads. Placing irrelevant ads above the search results damages user experience and should be avoided as much as possible. Likewise, placing too many ads on a page degrades overall user experience, particularly if low relevance ads are displayed.

3. LEARNING AD RELEVANCE MODELS

We learn a model of ad relevance that will allow us to use use predicted relevance to improve our sponsored search system. Our relevance model is a binary classifier trained to detect relevant and irrelevant advertisements, given a particular search term. We experimented with Maximum Entropy (maxent, i.e. [22]), adaBoost Decision Tree stumps (adaBoost, [31, 13]), and Gradient Boosting Decision Trees (GBDT, [37]). The baseline model had 19 features: query length plus 6x3 features that separately compared the query to the three zones of an ad (the title, description and display url). These six features included word overlap (unigram and bigram), character overlap (unigram and bigram), cosine similarity, and a feature that counted the number of bigrams in the query that had the order of the words preserved in the ad zone (ordered bigram overlap). This 19 feature model forms our baseline model, additional details are available in [26].

The target for our models was generated from editorial data on a five point editorial scale (Perfect, Excellent, Good, Fair, Bad), where we consider all judgments better than "Bad" as relevant and the remaining "Bad" judgments as

irrelevant ads. Judgments are performed by professional editors that achieve reasonable consistency. Our training set contains about 80k editorially judged query ad pairs. Our precision and recall results for detecting relevant ads are reported on an editorial test set of 40k query ad pairs. Training and test data were retrieved from our advertiser database with a TF-IDF based ad retrieval system similar to [6] (an average of 20 ads is retrieved per query). The data contains 7k unique queries, which were selected based on a stratified sample of search engine traffic that represents all ten search frequency deciles.

The results for three machine learning approaches are presented in Table 1. We compare to the baseline TF-IDF ad retrieval system, as well as a random baseline. The random baseline achieves the maximum F-Score by predicting all ads as relevant, which essentially results in precision based on the prior where about 20% of the test set is relevant. All models outperform the baseline TF-IDF system at their Max F-Score point, and adaBoost and GBDT are somewhat better than the maxent model. All pairwise differences between models (except adaBoost versus GBDT) are statistically significant with a binomial sign test (at $p < 0.01$). Figure 1 illustrates the precision recall curves. For the remainder of this work we build on the baseline GBDT relevance model, all model tuning parameters are optimized on a separate held-out set.

Model	Precision	Recall	Max F-Score
maxent	0.658	0.458	0.540
adaBoost	0.670	0.543	0.600
GBDT	0.671	0.551	0.605
TF-IDF	0.519	0.552	0.535
random	0.223	1.000	0.365

Table 1: Precision and Recall for various models on the relevance prediction task at the maximum F-score.

4. INCORPORATING USER CLICKS IN RELEVANCE MODELING

Our baseline relevance model is able to predict relevance with reasonable accuracy based on simple text overlap features, but it will fail to detect relevant ads if no syntactic overlap is present. An ad with the title "Find the best jogging shoes" could be very relevant to a user search "running gear" but our baseline model has no knowledge that running and jogging are highly related. Sections 4.1 and 4.2 introduce two approaches for leveraging user click data to learn semantic relationships between queries and ads.

4.1 Click History as Relevance Features

Historical click rates for a query-ad pair can provide a strong indication of relevance and can be used as features in our relevance model. User click rates often correspond well with editorial ratings when a sufficient number of clicks and impressions have been observed. The relationship is however not deterministic (as discussed earlier), so we allow the model to learn how to incorporate observed click rates. When there is not click history for a specific query-ad pair we can back off to higher levels that aggregate history across all ads in an adgroup, campaign, or an entire account. These aggregations benefit from observed click behavior on similar ads (from the same advertiser) and have been shown to provide significant gain in predicting click probability as described later in Section 5.3. We include multiple different types of aggregations that are already available to us from other parts of the sponsored search system.

While these click history features do provide important features, they are only available for a portion of the ads that has seen sufficient search traffic (this accounts for less than 10% of ads in our retrieved training set at the query-ad level, and 99% at the account level). Ads that are new to the system or occur for infrequent tail terms will not have reliable click history, so it is important to ensure that adding click features to our relevance model does not diminish model accuracy for these terms. We simulate the worst case scenario of no available click history by training the model with click features but then testing with the click features "blanked" out for all test examples.

Figure 2 presents the precision-recall curves for three models: the baseline text-only model, the model with text and click features, and the model trained with text and click features but tested with the click features "blanked" out. The "blanked" results indicate how the model with click features will perform compared to the baseline model when evaluating a new or infrequent ad that has no observed click history. Table 2 contains precision, recall, and max F-score for the models. All pairwise differences between results are statistically significant with a binomial sign test (at $p < 0.01$). The addition of historical click features provides a large improvement in precision compared to the baseline, although model precision is slightly degraded for the case of missing click history.

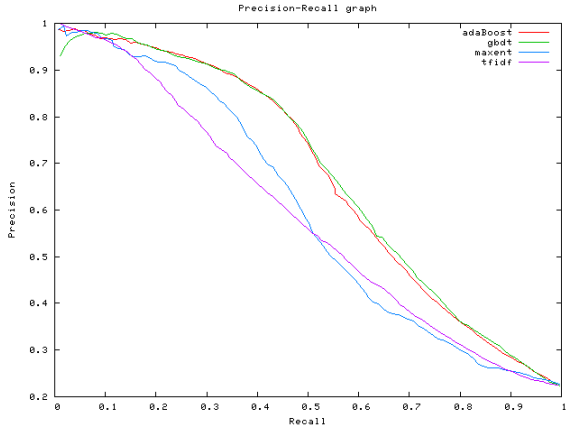

Figure 1: Precision/Recall curves for various relevance model learning approaches

4.2 Click Propensity in Query/Ad Translation

While the click features discussed in Section 4.1 are helpful for ads with sufficient click history, we could also use click information to learn relationships that are not tied to a particular ad or advertiser (as the current click history features are). Previous research has proposed modeling the query as a translation of the document for information retrieval [2], where the relevance of a document (in our case ad) and query can be modeled with Bayes' rule as:

Features	Precision	Recall	Max F-Score
baseline	0.671	0.551	0.605
+click	0.699	0.557	0.620
+blanked	0.652	0.556	0.600

Table 2: Precision and Recall for training with click history features on the relevance prediction task.

Model	Precision	Recall	Max F-Score
baseline	0.671	0.551	0.605
+trans. features	0.658	0.590	0.622
+click and trans.	0.673	0.584	0.625

Table 3: Results for baseline GBDT and models adding click likelihood translation score features and historical click features.

$$p(D|Q) = p(Q|D)p(D)/p(Q) \quad (2)$$

where $p(Q)$ can be ignored because it is constant for each particular query. The $p(Q|D)$ term can be considered a statistical translation problem and decomposed using IBM Model 1 [8] in the form:

$$p(Q|D) = \prod_{j=0}^{m} \sum_{i=0}^{n} trans(q_j|d_i) \quad (3)$$

for query words $q_0...q_m$ and document words $d_0...d_n$ where $trans(q_i|d_j)$ is a probability of co-occurrence collected over some corpus of parallel queries and documents. The maximum likelihood estimations of the co-occurrence statistics are normalized counts over the training corpus (in our case the ad click logs, $logs$):

$$trans(q_j|d_i) = \frac{\sum_{logs} count(q_j|d_i)}{\sum_q \sum_{logs} count(q|d_i)} \quad (4)$$

The translation probability counts the number of clicks a query-ad word pair received, divided by the total number of clicks that ad word received across all query words. The *count* function can also be updated with EM iterations, where the $trans(q_i|d_j)$ from the previous iteration weights the co-occurrence counts. In our case we additionally smooth the counts with generalized absolute discounting described in [35]. Finally, the $p(D)$ of Equation 2 can be represented as a unigram language model, multiplying the probabilities of the document (ad) words that are also collected from the smoothed counts on the click logs.

We learn two translation models, where the first simply takes the number of clicks as the co-occurrence counts. We then train a second model using statistics collected over all query-ad pair impressions in the logs. Impressions are weighted by "expected clicks" (ec) based on a rank normalization [36, 10]. For an ad a at rank r that has been retrieved for a query q, we define ec as:

$$\text{ec(q,a)} = \sum_r \text{imp}(q, a, r) P(\text{click}|r) \quad (5)$$

The quantity $\text{ec}(q, a)$ is the expected number of clicks summed over all rank positions that an ad appears in. The quantity $P(\text{click}|r)$ is estimated by observing the per-position click-through rate on a size-able portion of search traffic for several days.

We can then take a ratio of the translation probability from the click counts, $p_{click}(Q|D)$, divided by the probability from the expected click counts, $p_{ec}(Q|D)$, to determine a click propensity:

$$clickLikelihood = \frac{p_{click}(Q|D)}{p_{ec}(Q|D)} \quad (6)$$

This likelihood ratio, or click propensity, provides a score that removes the presentation bias from the log based translation models. The $p_{click}(Q|D)$ translation model, based only on clicks, can be biased because a strong click signal may appear from even a low click rate on a massive number of impressions. The above likelihood ratio divides by the probability of click that would be expected on average from the weighted impressions, so query-ad pair will have a large ratio when it gets more clicks than would be expected from average term pairs.

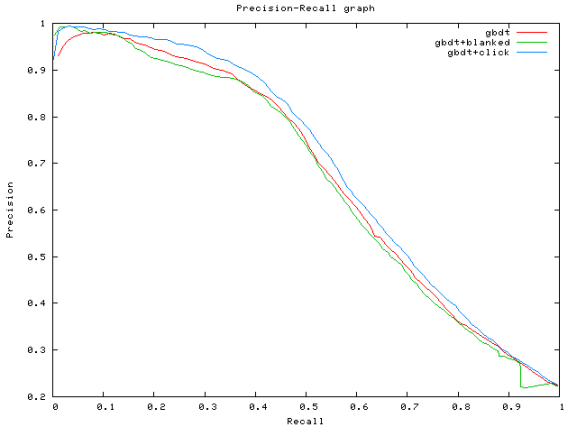

Figure 2: Precision/Recall curves when including historical click features

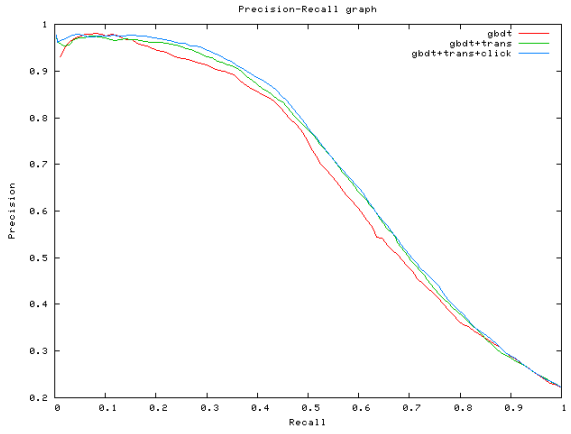

Figure 3: Precision/Recall curves when including click likelihood translation score features and historical click features.

5. SPONSORED SEARCH APPLICATIONS

We include a translation likelihood score during relevance model training for both translation directions, from query to ad, and ad to query. Table 3 compares our baseline GBDT model to a model trained with the translation click likelihood scores as additional features in the model, as well as combining with the click history features from Section 4.1. The translation scores provide a large recall improvement (7% relative) with some reduction in precision (2% relative). Combining with observed click history features recovers precision while maintaining the improved recall. All pairwise differences between results are statistically significant with a binomial sign test (at $p < 0.01$).

Figure 3 shows precision-recall curves for including translation features and historical click features. The addition of translation features provides a similar improvement when comparing to the direct click history features. The translation scores have the additional benefit that the features generalize to query-ad pairs that do not have any click history because the translation score is based solely on the ad text (whereas observed click rates typically depend on a specific ad or advertiser). When translation features are combined with the direct historical click features we obtain a further improvement, particularly in the higher precision region.

Section 3 introduced a baseline relevance model and Section 4 developed improvements utilizing user clicks as features for relevance modeling. This section briefly describes our approach to evaluating our sponsored search system and then reports on experiments for three applications of leveraging predicted relevance to improve sponsored search.

5.1 System Evaluation

Evaluation of a live search system can be complicated and ambiguous. The large amount of user traffic means that human judgments are intractable for any significant portion of the data. Alternatively, statistics collected over millions of user interactions are available and can provide significant insights into the impact of an experimental approach (but logs can include noise from spam and other sources). A complete analysis typically incorporates editorial judgments by humans over a small sample of the data, combined with measurements of user behavior such as click rates. We have access to a platform that allows us to run our experimental system live on a fraction (or "bucket") of traffic for a large commercial search engine.

We can estimate measures that indicate the quality of the individual systems by analyzing the logs for a large number of page-views generated from our experimental system and comparing to a baseline system. Additionally, because in "pay-per-click" advertising the desired goal of the search engine is user-clicks on ads, metrics from "bucket-testing" can help evaluate the monetization capability of the new algorithm. For an introduction to bucket testing the reader is referred to the paper by Kohavi et al [18]. In this paper we report bucket test results for experiments in three sponsored search applications. The relevance model deployed in the buckets does not use the translation model described in section 4.2 for reasons of computational efficiency and latency.

5.2 Filtering low quality ads

As described in Section 2, the goal of the initial stage of most sponsored search systems is to retrieve a candidate set of relevant ads for a particular search query. The set of candidate ads is a pool generated by various retrieval technologies that rely on query rewriting methods as well as direct ad retrieval such as the approaches described in [1]. In order to improve the relevance of the final candidate set we will apply our relevance model to each query-ad pair in the candidate set and prune those ads that do not meet a relevance threshold. Table 4 presents the results of a live bucket test that applies the relevance model online to all candidate ads, removing those ads that do not meet a relevance threshold.

Metric	Relative Change
Clicks per candidate set (CTR)	+10.1%
Queries with ads (coverage)	-8.7%
Ads per query (depth)	-11.9%
Clicks per search (Click Yield)	+0.5%

Table 4: Bucket metrics for relevance filtering.

Editorial Rating	Total Ads	Filtration Rate
Perfect	267	3% (8)
Excellent	206	8% (17)
Good	3888	8% (306)
Fair	10912	17% (1849)
Bad	8526	49% (4215)

Table 5: Filtration rates per editorial category.

The filter significantly reduced the number of ads displayed to the user, with an 8.7% reduction in queries with ads (coverage) and an 11.9% reduction in the average number of ads per search query. Even with this large reduction in the number of ads the average clicks per search was neutral to slightly positive, which indicates most all of the removed ads received few clicks in the baseline production bucket. Cutting the number of ads shown while maintaining constant total clicks is also indicated by the 10% increase in click through rate.

While click metrics give some indication of how well our relevance model filtering is performing, we are primarily interested in reducing low relevance ads. An online test has too many events to measure everything editorially, so we also sampled five thousand random queries from the bucket and gathered an editorial evaluation of the query ad pairs. Applying our relevance filter to this set we can determine what percentage of ads are filtered for each editorial grade. Table 5 illustrates that our filter eliminates about half of all bad ads, with marginal impact on the higher quality ads. The combined impact is a significant reduction in total ads (identified editorially as primarily low relevance) along with maintaining constant to positive overall clicks, which together should indicate an improved user experience.

5.3 Ranking ads with low click history

As noted in Section 2, ads with little observed click history are difficult to rank by probability of click. In this section we incorporate the predicted ad relevance as a feature in ranking with the intention of improving click prediction (particularly when little click history is available). Ads are ranked by a machine learned model that predicts the probability that the user is likely to click on an ad for a query,

$p(click|query, ad)$. We learn a maximum entropy model for this task, which has the following functional form:

$$p(click|query, ad) = \frac{1}{1 + exp(\sum_i w_i f_i)} \quad (7)$$

where f_i denotes a feature based on either the query, the ad, or both and w_i is the weight associated with the feature. The model is described in more detail in the work of Shaparenko et al. [33] and is learned from the query click logs. Each line in the query log contains a query and an ad, whether the ad was clicked, and other information such as the time-stamp and the position on the page that the ad was shown to a particular user. This data is used to train a binary classifier using the maximum entropy model as described above.

While any supervised classification algorithm may have been used (eg., [29, 11]) the learning framework for maximum entropy has been efficiently parallelized using Hadoop to handle billions of training samples. Maximum entropy models can also handle sparse and mutually correlated feature-sets reasonably well. The primary features for the model are various levels of historical click aggregation, which are supplemented by additional features such as time of day, as well as some simple syntactic overlap features which are a similar to those used in the relevance model described in Section 3.

The ranking model is typically evaluated offline with analysis of precision/recall curves, where the test events are hundreds of millions of click and non-click events from the search logs. Model performance is very accurate when sufficient click history is available (because future clicks track past click behavior very closely). The task is more challenging when little or no click history is available for an ad, such as the case of a newly created ad or an ad for a very infrequent tail search. This problem can be partially approached by inferring click history information from the ad database hierarchy, such as the average click history for all of the ads from a particular account. But, when absolutely no click history is available the model must predict the probability of a click given the remaining features, such as overlap between query and display url or query and ad title.

Our baseline relevance model can predict relevance independent of click history, so we can include the relevance score as an additional feature input to the model. We compare the performance of the baseline click ranking model with a ranking model that incorporates the predicted relevance as a feature. Figure 4 evaluates events that had no account level click history (1.2% of test data), and Figure 5 evaluates events with no adgroup level history (3.6% of test data). We find that including the relevance score as a feature provides useful gains in both settings for the high precision regions, indicating that the top ranking ads are ranked more accurately. Precision for these ads improves by more than 20% (relative) for events with no adgroup level history. The "area under the curve" improved by 3.5% (relative) for events with no adgroup history, and improved by 5% (relative) for events with no account history. The results on the remainder of the test set, where sufficient click history is available, were unchanged from the baseline click ranking model.

	Click history levels		
	None	Low History	High History
Rank1 CTR	+0.1%	+12.7% ($p < 0.05$)	-0.5%
Rank2 CTR	+2.8%	+16.9% ($p < 0.1$)	+1.3%

Table 6: Bucket results for click ranking.

We also conducted a bucket test to compare the click model that uses the relevance model score as feature to the baseline model. Table 6 compares the CTRs of these two models on query slices with varying click history aggregates. We present the results for only the top 2 positions because the sample sizes for the lower positions were not big enough to show any significant changes. The results are presented per position so that there is no presentation bias in the resulting click rates.

The model that includes predicted relevance as a feature has increased both rank 1 and rank 2 CTRs significantly, by 12.74% and 16.68% respectively, for the low query history slice. We did not observe any changes for the queries that already had sufficient historical information. Note that these findings were predicted by our offline analysis presented above. Surprisingly, we did not see the improvements that were predicted for the query slice with no history at all. This may be due in part to the small sample size collected

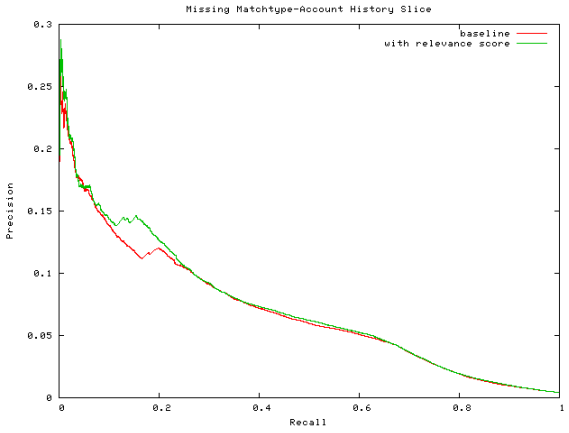

Figure 4: Click prediction precision/recall curve for events with no account level click history.

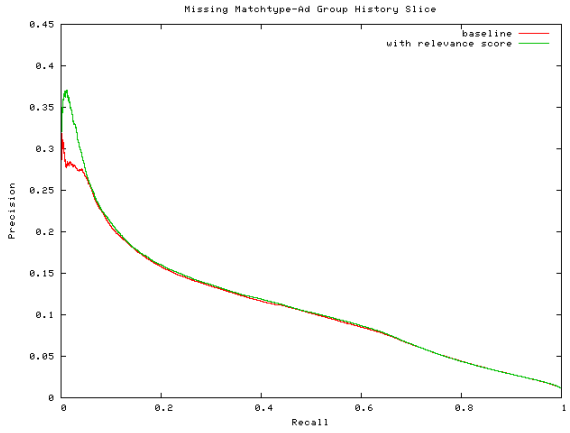

Figure 5: Click prediction precision/recall curve for events with no adgroup level click history.

for this set of queries and in part to the nature of the queries themselves. The absolute number of clicks for these queries are comparatively much smaller, suggesting that either these queries are not commercial or that the ad set was not attractive enough to begin with.

5.4 Reducing North Ad Impact

Given a ranked set of candidate ads, the final stage of sponsored search should decide how many ads to place in the north above the organic search results. Placing advertisements on top of the organic search results (rather than to the side in the east) creates a direct competition between ads and search results. For commercial search terms ads can be more attractive than web results. More frequently, however, they can divert the user's attention and might keep them from ultimately reaching pages containing the information they requested. The search engine can deliberately incur degradation of user experience in exchange for expected revenue. Ads not shown in the north can still be shown in the east or in the south; however, the bulk of both user experience impact and revenue stems from north ads because of their prominent position on the page. One way of measuring search retrieval quality that has become somewhat of a standard is the Discounted Cumulative Gain (DCG) [15]. This is a weighted sum of the editorial relevance (according to human judges) of the top returned documents, where the weight is a decreasing function of the rank:

$$DCG_n = \sum_{i=1}^{n} w_i \cdot rel_i \tag{8}$$

This formula is typically used with graded relevance scores, and weights that place much more importance on higher ranks (we use $1/log_2(rank + 1)$). When ads placed above the search results degrade overall quality the degradation can be measured as North Ad Impact (NAI), the percent decrease in DCG introduced by displaying ads:

$$NAI = \frac{DCG_{noAds} - DCG_{withAds}}{DCG_{noAds}} \tag{9}$$

The DCG_{noAds} computes DCG over the top five organic search results, while $DCG_{withAds}$ computes DCG over the top five results including ads (for instance with 3 north ads DCG is computed over the 3 ads and the top two organic search results). We can attempt to reduce NAI in our sponsored search system by estimating DCG before and after potential north ad placements and choosing to place ads in the north where we incur the lowest NAI penalty (generally when ad relevance is higher and web relevance is lower). The ad DCG score is estimated with our relevance model and the search engine ranking score estimates the organic search DCG score.

The baseline ad relevance model was trained for the binary relevant vs. irrelevant task, but for page placement we desire a prediction of the editorial relevance score that is a five point non-linear scale. Therefore we retrain the relevance model as in Section 4.1, but using GBDT regression on the corresponding editorial point value for each grade as the target, rather than a binary good versus bad target (so as to predict on the same scale as the estimated organic result relevance). Finally, we can estimate the NAI of placing any ad in the north by comparing the predicted ad and web relevance scores.

Metric	Relative Change
Editorial NAI	-4.5%
North Click Through Rate	+1.7%
Click Yield	+.8%

Table 7: Bucket metrics using estimated NAI in north ad placement.

Sponsored search is typically allocated a fixed number of average north ads per query (based on revenue considerations) and then chooses when to place ads in the north for a particular query by optimizing some criteria. Our baseline system optimizes north ad placement based on a combination of maximum revenue (the probability of click times the advertiser bid) and user cost (a penalty for low quality ads). Ads are ordered by the optimization function and are then placed in the north if their score is above a threshold that produces an average quota of north ads. Table 7 presents bucket results for an alternative page placement strategy that incorporates our predicted NAI impact as the user cost, where the baseline system uses a function of the estimated $p(click|query, ad)$.

By incorporating an estimate of NAI directly in the optimization function we are able to place the same quantity of ads in the north, but with a lower impact on the user because we bias towards higher relevance ads (and take the web result relevance into consideration). The bucket results show good improvement (reduction) in NAI along with a corresponding increase in north ad CTR and click yield, which are additional measures of user satisfaction. The NAI results are calculated based on editorial assessment over one thousand randomly sampled queries, for which both north ads and organic search results are judged.

6. RELATED WORK

Work in online advertising focuses on two main areas: contextual advertising and sponsored search. Sponsored search has been described in detail in section 2. Contextual advertising is a similar problem that mainly concerns itself with the placement of ads on publisher pages, such as news pages or blogs. Research in these two areas is related to our work, as are many topics in traditional information retrieval.

Several methods that use supervised learning techniques with data labeled on an ordinal scale to learn a classifier or a ranking function have been proposed (e.g., [21, 9]). In sponsored search however, most of the approaches published in the literature so far have either taken a traditional information retrieval approach (e.g., [7]) or one that learns a classifier based on click data (e.g., [29, 11, 32]). To our knowledge this work is the first to show the benefit of modeling human-assessed relevance for many tasks in sponsored search. We find that our model can improve performance compared to a baseline TF-IDF framework. We also find that modeling human-judged relevance can even improve a classifier that predicts click-through-rate, especially on the slice of traffic for which little historical impression data exists. Finally, we find that our model can be used to filter low quality ads and to reduce North Ad Impact, resulting in improved user experience.

While filtering has been well studied in information retrieval [20], little work has been done in the context of web-

search and ads in particular. Perhaps, the most closely related work is that of Broder et al. [4] who trained a classifier for a similar problem: given a query and a slate of ads, they predicted whether or not to show advertisements for the query. Our classifier on the other hand predicts whether or not to show the ad for a query-ad pair. In comparison with Broder et al[4], our work explores several models and our evaluation is on a data-set that is several times larger, and we also evaluate on live traffic.

In contextual advertising publisher pages are rich in content and a rich set of features can typically be extracted from the web-page and used to find relevant ads [5, 34]. The sponsored search problem on the other hand suffers from the same problem as web-search, where the queries are short and have little context. Exacerbating the problem is the fact that the ad document is short with little context as well. One way of overcoming this problem is though "query rewriting" techniques. The transformed query is then used for ad retrieval. Models to predict query rewriting techniques may be learned from query logs [17, 25]. An alternate way of overcoming the issues posed by matching documents to short queries is query expansion, a technique well studied in information retrieval [30, 19]. Ribeiro-Neto et al. [28] found expanding the content of publisher pages to be useful to the problem of contextual advertising. Likewise, query expansion has been found to be beneficial in sponsored search [7, 27]. Incorporating topic clusters or hierarchical categorical features has also provided improvements [32].

Many systematic frameworks for query expansion exist in information retrieval. In this paper we chose the translation models of Berger and Lafferty [2] since several recent works [16, 23] have found this framework useful for tasks where the documents to be retrieved are very short. While Jeon was attempting a Q&A retrieval task, Murdock was attempting a sentence retrieval task. Murdock et al. [24] also applied the translation model approach to contextual advertising. They computed translation models between a small set of publisher pages and landing pages by using a parallel corpus determined by human judgments.

We presented a variation of the translation models in which the translations are trained using billions of events from our click logs. Our translation models can easily be automatically computed from search engine logs and therefore our method is more robust to seasonal variations in the vocabulary of commercial terms (and can be collected over corpuses many orders of magnitude larger than most previous work). A similar method was proposed by Raghavan and Iyer [27]. However, their translation model was a simple list of co-occurring words in clicked query-ad pairs. In contrast our method uses click frequency while normalizing by position normalized impressions. Other related work has found gains from incorporating word pair features directly as features in a sponsored search click prediction model [33].

7. CONCLUSIONS

We have presented a baseline relevance model that accurately predicts relevance for query-ad pairs, and additionally improved that model by incorporating implicit relevance feedback from sparse user clicks in search logs. We found that observed click history is helpful in predicting relevance when sufficient observations are available. When few or no observations are available, we described a method for learning a translation model from click logs that can generalize to unseen ads by relying on the ad text. Both approaches provide significant improvements to the task of predicting relevance.

We then applied our relevance model to three major components of a sponsored search system: ad retrieval, ad ranking, and page placement. Both offline editorial evaluation and online bucket metrics show significant relevance improvements across all three tasks. Future work could extend estimation of translation models to include text from organic web results, as well as incorporating topical and categorical features.

8. REFERENCES

[1] T. Anastasakos, D. Hillard, S. Kshetramade, and H. Raghavan. A collaborative filtering approach to ad recommendation using the query ad click graph. In *CIKM*, 2009.

[2] A. Berger and J. Lafferty. Information retrieval as statistical translation. In *SIGIR*, 1999.

[3] A. Broder. A taxonomy of web search. *SIGIR Forum*, 36(2), 2002.

[4] A. Broder, M. Ciaramita, M. Fontoura, E. Gabrilovich, V. Josifovski, D. Metzler, V. Murdock, and V. Plachouras. To swing or not to swing: learning when (not) to advertise. In *CIKM*, pages 1003–1012, 2008.

[5] A. Broder, M. Fontoura, V. Josifovski, and L. Riedel. A semantic approach to contextual advertising. In *SIGIR*, 2007.

[6] A. Z. Broder, D. Carmel, M. Herscovici, A. Soffer, and J. Zien. Efficient query evaluation using a two-level retrieval process. In *CIKM*. ACM, 2003.

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

[8] P. F. Brown, V. J. Pietra, S. A. D. Pietra, and R. L. Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19:263–311, 1993.

[9] C. J. C. Burges, R. Ragno, and Q. V. Le. Learning to rank with nonsmooth cost functions. In *NIPS*, pages 193–200. MIT Press, 2006.

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

[11] M. Ciaramita, V. Murdock, and V. Plachouras. Online learning from click data for sponsored search. In *WWW*, 2008.

[12] B. Edelman, M. Ostrovsky, and M. Schwarz. Internet advertising and the generalized second-price auction: Selling billions of dollars worth of keywords. *American Economic Review*, 97(1), March 2007.

[13] B. Favre, D. Hakkani-Tür, and S. Cuendet. icsiboost. http://code.google.com/p/icsiboost/.

[14] B. Jansen and M. Resnick. Examining searcher perceptions of and interactions with sponsored results. In *Workshop on Sponsored Search Auctions*, 2005.

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

[16] J. Jeon. *Searching Question and Answer Archives.*

PhD thesis, University of Massachusetts, Amherst, 2007.

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

[18] R. Kohavi, R. M. Henne, and D. Sommerfield. Practical guide to controlled experiments on the web: listen to your customers not to the hippo. In *KDD*, 2007.

[19] V. Lavrenko and W. B. Croft. Relevance based language models. In *SIGIR*, 2001.

[20] D. D. Lewis. The trec-5 filtering track. In *The Fifth Text REtrieval Conference (TREC-5*, pages 75–96, 1997.

[21] P. Li, C. J. C. Burges, and Q. Wu. Mcrank: Learning to rank using multiple classification and gradient boosting. In *NIPS*, 2007.

[22] T. Minka. A comparison of numerical optimizers for logistic regression. Technical report, Microsoft, 2003.

[23] V. Murdock. *Aspects of Sentence Retrieval*. PhD thesis, University of Massachusetts, Amherst, 2007.

[24] V. Murdock, M. Ciaramita, and V. Plachouras. A noisy-channel approach to contextual advertising. In *ADKDD '07: Workshop on Data mining and audience intelligence for advertising*, 2007.

[25] 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.

[26] H. Raghavan and D. Hillard. A relevance model based filter for improving ad quality. In *SIGIR*, pages 762–763, New York, NY, USA, 2009.

[27] H. Raghavan and R. Iyer. Evaluating vector-space and probabilistic models for query to ad matching. In *SIGIR '08 Workshop on Information Retrieval in Advertising (IRA)*, 2008.

[28] B. Ribeiro-Neto, M. Cristo, P. B. Golgher, and E. S. de Moura. Impedance coupling in content-targeted advertising. In *SIGIR*, 2005.

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

[30] J. Rocchio. *Relevance Feedback in Information Retrieval*. Prentice-Hall, 1971.

[31] R. E. Schapire and Y. Singer. Boostexter: A boosting-based system for text categorization. *Machine Learning*, 39(2/3):135–168, 2000.

[32] D. Sculley, R. G. Malkin, S. Basu, and R. J. Bayardo. Predicting bounce rates in sponsored search advertisements. In *KDD*, pages 1325–1334, 2009.

[33] B. Shaparenko, O. Cetin, and R. Iyer. Data driven text features for sponsored search click prediction. In *AdKDD Workshop*, 2009.

[34] W. tau Yih, J. Goodman, and V. R. Carvalho. Finding advertising keywords on web pages. In *WWW*, 2006.

[35] R. Zens and H. Ney. Improvements in phrase-based statistical machine translation. In *In Proc. of the Human Language Technology Conf. (HLT-NAACL*, pages 257–264, 2004.

[36] 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.

[37] Z. Zheng, H. Zha, T. Zhang, O. Chapelle, K. Chen, and G. Sun. A general boosting method and its application to learning ranking functions for web search. In *NIPS*, pages 1697–1704, 2008.