Data-Driven Text Features for Sponsored Search Click Prediction

Benyah Shaparenko
Cornell University
4130 Upson Hall
Ithaca NY 14853
benyah@cs.cornell.edu

Özgür Çetin
Yahoo! Labs
111 West 40th
New York NY 10018
ocetin@yahoo-inc.com

Rukmini Iyer
Yahoo! Labs
4401 Great America Parkway
Santa Clara CA 95054
riyer@yahoo-inc.com

ABSTRACT

Much search engine revenue comes from sponsored search ads displayed with algorithmic search results. To maximize revenue, it is essential to choose a good slate of ads for each query, requiring accurate prediction of whether or not users will click on an ad. Click prediction is relatively easy for the ads that have been displayed many times, and have significant click history, but in the long tail with minimal or no click history, other features are needed to predict user response. In this work, we investigate the use of novel text features for this problem, within the context of a state-of-the-art sponsored search system. In particular, we propose the use of detailed word-pair indicator features between the query and ad. We compare the new features to the traditional vector-space and language modeling features extracted in a typical information-retrieval style. We evaluate these approaches in a maximum-entropy ranking model using the click-view data from a commercial search-engine traffic. We show that the word-pair features are highly helpful for sponsored search click prediction, not only improving over the sophisticated click-history feedback based systems, but also compensating for the lack of click history to some extent. In contrast, we find that the language and vector-space modeling approaches are significantly less effective.

Categories and Subject Descriptors

H.3.5 [**Online Information Services**]: Commercial Services; I.5.2 [**Design Methodology**]: Classifier design and evaluation; H.3.3 [**Information Search and Retrieval**]: Relevance feedback

Keywords

Sponsored search, click prediction, text features, maximum entropy model, language model, vector-space model

1. INTRODUCTION

ADKDD'09, June 28, 2009, Paris, France.

Internet search engines derive a large portion of their revenue from the sponsored search ads that appear on the search results page. Advertisers can place their ads by making bids on search keywords, called the bidded keywords. When the user inputs a query, the search engine not only returns a set of relevant search results, but also ads that are potentially interesting to the user. The search engines typically use a pay-per-click model in which they receive revenue each time the user clicks on an ad. The advertiser hopes to convert the user's click-through to its site into revenue, while the search engine tries to maximize its revenue. To maximize revenue, the ads are typically ranked by the expected revenue (which is equal to bid times the probability of click) in a second-price auction; see, e.g., [11]. The probability of click is the likelihood that the user will click on the ad if the ad is displayed for that query. It is unknown, and its estimation is the central problem in sponsored search. An accurate estimate would allow the search engine to display the most relevant ads, and price them correctly in an auction. Given the scale of search traffic, small errors in finding this probability can result in much lost revenue, and adverse user experience.

There are dynamic and static information sources that can be used to predict click. For the ads that are frequently displayed, there may be enough click history so that a feature such as click-through rate can indicate how the users behave when faced with a particular query, an ad, and ideally, a query-ad pair. For the ads without much history, the text sources such as the query, bidded keywords, ad title, and ad abstract can be crucial for finding syntactic and semantic clues that can indicate how closely the ad is related to the query. This paper is concerned with the utility of such text features for sponsored search click prediction. Since the click history has a big impact on the performance, we study the effect of the new features with and without click history.

We explore two approaches for using text for sponsored search click prediction. These approaches differ in the amount of information processing done before the final ranking model. In the first approach, advocated in this paper, almost no processing is performed: the words themselves are used as features in the ranking model. In the second approach, we use various scores extracted using vector-space and language models in a traditional information-retrieval (IR) style. There are trade-offs between these two approaches. First, the word-based approach is highly flexible, data driven, and lets the machine-learning model figure out the relevant information, while the score-based approach

aims at summarizing the information in a small number number of features. Second, the word-based approach has potentially many more parameters that need to be learned from data than the second approach. Therefore, scalable learning algorithms are a must for it to work. Third, the word-based approach can be less robust to changes in content between training and testing. This paper addresses some of these issues, and shows that given the large amounts of the click-view data available in sponsored search, the data-driven approach based on words is feasible, scalable, and more effective than the score-based approaches, including language and vector-space modeling.

The main contributions of this paper are the introduction of novel word-pair indicator features for sponsored search prediction, and extensive comparisons between those features and the traditional vector-space and language-modeling features, within the context of a commercial system. The paper is organized as follows. Section 2 describes the sponsored search problem and the baseline system. Section 3 presents various text features, including the word-pair indicator features. Section 4 presents experimental results on the click-view data. After a discussion of the results and some related work, we draw several conclusions, and give insights for future work.

2. CLICK PREDICTION

We approach to the sponsored search click prediction problem in a supervised learning paradigm. For each ad a put in front of the user for the query q, we record the user's response c (1 for click and 0 otherwise), and use this data to estimate a model for the probability of click $p(c|q,a)$. Many models can be used for this purpose, e.g., [29, 5, 3], but in this work, we use a maximum-entropy (ME) model mainly because the ME model can handle large, sparse, overlapping feature sets well. In addition, there are efficient, parallelizable algorithms for learning the ME model from data. The ME model, also known as the logistic regression, takes the following form:

$$p(c|q,a) = \frac{1}{1+\exp(\sum_{i=1}^{N} w_i f_i)} \tag{1}$$

where f_i denotes the i-th feature, w_i the associated weight, and N the total number of features.

The features can be derived using any information available in the context. As mentioned previously, we extract features from both dynamic and static sources. The most important dynamic features are the click-through rate features, which can be highly informative if enough history is available. These history features can be extracted in a hierarchy to provide coverage for the new queries and ads, or more generally, share information from related ads or queries, e.g., [29]. Some form of rank normalization is usually necessary to remove position bias, e.g., [37]. The static sources include the query and ad texts, which is the focus of this paper. While these texts are likely to be useful with or without click history, we expect them to be especially useful in the long tail with little or no click history. We study the effectiveness of the text with respect to varying amounts of click history available. There are other sources such as user's online history that can be utilized for click prediction.

ME modeling involves two key problems: the estimation of weights w_i, and the design of features f_i. Given a training data set, the first problem can be solved by maximum likelihood estimation, which is well-studied for the ME models, e.g., [23]. In this work, we use a custom nonlinear conjugate-gradients algorithm, which can handle large feature sets, as we will explore with the word-pair features. In addition, we use a Gaussian prior over the weights for regularization. Feature design, on the other hand, is a more difficult and domain-specific problem. It can be computationally intensive, and a scoring function can be helpful to rank features and eliminate the least useful ones. We use the cross-correlation with the click for this purpose [10]:

$$\rho_{CS} \equiv \frac{\sum_t (c_t - \bar{c})(s_t - \bar{s})}{\left(\sum_i (c_t - \bar{c})^2\right)^{1/2} \left(\sum_t (s_t - \bar{s})^2\right)^{1/2}} \tag{2}$$

where c_t and s_t denote the click indicator and feature value, respectively, and $\bar{c}$ and $\bar{s}$ the corresponding sample means. We note that while the cross-correlation may not be ideal for discrete variables, we found it to be a computationally convenient and effective metric.

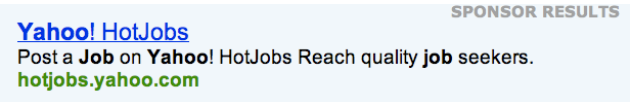

Figure 1: A sponsored search ad for the query *yahoo hot jobs*. The first line is the title, the second the abstract, and the third the display URL.

3. TEXT FEATURES

Since the sponsored search ads exclusively are textual, the terms or words in the ad, and their relation to the user's query should give some indication of whether the user will click on the ad (there are other factors such as the advertiser reputation, and the prior user experience). The three sources of ad texts that we use in this work are bidded keywords, ad title, and ad abstract. The bidded keywords are the search terms that the advertiser wants to target for matching queries. The matching is exact if the bidded keywords are completely included in the query. It is advanced if the ads can be matched against queries that do not have to include the bidded keywords, but are found to be relevant using other sources of information, for example, user query rewrites [33, 36, 27]. The bidded keywords are not visible to the user, only the title and abstract (see Figure 1). Title contains a short caption, and abstract a summary of the product or service. It can be argued that while the bidded keywords, title, and abstract are in increasing order of the information they contain about the underlying ad, that information is also spread across more and more words. Any of these fields can be untruthful. Whether or not the extracted features are vulnerable to spam is an important concern in sponsored search.

In this section, we present three textual feature extraction methods for sponsored search click prediction in the ME modeling framework. The first two methods, vector-space and language modeling, are well-known in the IR community. They summarize any information that might be present in the text into a small number of scores; the words are ignored afterwards. The third method, word-pair indicator features, is the main contribution of this work, and

employs the words themselves as features in the ME model. These features allow the final prediction model to induce relevant features in a data-driven fashion with parameters learned from the data.

3.1 Vector-Space Model

The vector-space model is one of the most widely used models for unstructured text data [31, 30]. The vector-space model represents each document as a vector with dimensions corresponding to separate terms, and weights emphasizing content terms. In term frequency-inverse document frequency weighting, the i-th dimension of the document vector v is given by

$$v_i \equiv \frac{f_i}{|D|} \log \frac{|C|+1}{n_i+0.5}$$

where f_i denotes the term frequency, $|D|$ the document length, $|C|$ the collection size, and n_i the document frequency of the i-th term. Using this model, a measure of similarity is the cosine of the angle θ between the query w^q and ad w^a vectors,

$$\cos(\theta) = \frac{v^q \cdot v^a}{|v^q||v^a|}. \quad (3)$$

The ad vector can be constructed using bidded keywords, title, or abstract, and the cosine similarity can be used as a feature in the ME, or some other model [24, 27].

3.2 Language Model

The language modeling approach to IR ranks documents according to how likely it is to generate the search query using each document as the source model. Given a query q, a document d is ranked by the probability $p(q|d,r)$, where r denotes relevancy. In the sponsored search context, we take ads to be documents, and limit ourselves to the clicked ads. Thus, we assume that click equals to relevancy, which is reasonable for the click-view data. This assumption additionally removes the need to use editorial data, or take the position effects into account. If the user clicked the ad, we assume that it was viewed and considered.

We use the standard translation-based language model to calculate the likelihood of the query $q = (q_1, \ldots, q_L)$:

$$p(q|a,r) = \prod_{i=1}^{L} \sum_{w \in V} p(q_i|w,r)\, p(w|a) \quad (4)$$

where V is the vocabulary, $p(q_i|w,r)$ the translation table, and $p(w|a)$ the ad model [1, 15]. Notice that Equation (4) entails the unigram assumption in which each query word is modeled independently of other query words. The translation model $p(q_i|w,r)$ allows for interactions between the query and ad words that go beyond the exact match, and can be calculated using the clicked query-ad pairs [27]. The document model $p(w|a)$ is constructed by interpolating the maximum-likelihood ad model with the corpus ($\mathcal{C}$) model:

$$p(w|a) = \lambda\, p_{ml}(w|a) + (1-\lambda)\, p(w|\mathcal{C}).$$

We experimented with λ values of 0.1, 0.5, and 0.9 to explore trading off document specifity for probability smoothing.

We use the model in Equation (4) to derive a number of features. The first two approaches below concern with the particular model used, and the last two with the transformation of the resulting probabilities before feeding them into the ME model.

1. **Identity-Translation Model**. In the simplest approach, the language-modeling probabilities are directly used in the ME model without any modification. In addition, a trivial, identity translation model is used. This model cannot account for any relationships beyond exact syntactic match, and the scores are analogous to the cosine distance in Equation (3).

2. **Full-Translation Model**. Next, the full translations are included. The translation model is estimated using the clicked ads in the training data (extra information sources such as web search results might also be helpful, but see Section 3.3). Because ranging over the entire vocabulary in the translation model introduces many noisy terms that happen to just co-occur, we use only the 20 most likely ad terms associated each query term, which has the side effect of making the computation feasible.

3. **Query-Length Normalization**. One of the potential problems with the language-modeling probabilities is that they are in general smaller for longer queries due to the unigram assumption. The length normalization can be performed in two ways. In the first approach, we normalize the probabilities by the query length in the log domain (the resulting scores are essentially the same as the perplexity metric used in speech recognition [4]). In the second approach, we factor the query length in the ME model, so that a separate weight in Equation (1) is used for probabilities coming from queries of each length.

4. **Score Binning**. According to the ME model, the relationship between the click probability and features is log-linear and monotonic. To allow for nonlinear or non-monotonic relationships, and also introduce additional degrees of freedom, we applied the standard method of score binning to the language-modeling scores (again per query length). We investigate the monotonicity assumption in Section 4.4.

Any of these methods can be employed with the bidded keywords, title, or abstract. In general, we expect the longer the text, the more reliable the language-modeling scores.

3.3 Word-Pair Indicator Features

In both vector-space and language modeling, any information that might be useful for click prediction is conveyed into a handful of scores, which are then fed into the ME model. This approach significantly reduces the feature dimensionality and model complexity, but any information that is lost during feature reduction cannot be recovered. In addition, while the information captured by the vector-space and language modeling approaches is intuitive, and has been used in other IR tasks before, it might not be optimal for a particular ranking problem, or couple well with a particular model such as the ME model. Ideally, we would like to learn features directly from the data, according to how useful they are in the ME model. Such a data-driven approach can be especially helpful with the click-view data, which is noisy due to spam and accidental clicks, and implicitly labeled only [17, 18]. The noisy data implies a significant departure from the traditional IR learning paradigm.

Our approach to discovering salient syntactic and semantic relationships is to present the text data in a form as raw

Diagonal Word Pairs		Off-diagonal Word Pairs		Query-Term Absence Words	
ebay	hsn	ebay exactly	com com	njmv	armaniexchange
com	penney	ebay want	city $24	buycostumes	clubpogo
circuit	orbitz	ebay today	circuit $24	adultbouncer	cherylscookies
kmart	netflix	ebay official	circuit orders	audltfriendfinder	thecontour
expedia	overstock	ebay find	circuit shipping	scdmv	ajwright
macys	circuitcity	ebay site	city orders	magiccabin	firsttuesday
singlesnet	nordstrom	ebay ebay	city shipping	medifast	2crazyfox
kohls	adultfriendfinder	com official	circuit online	sharebuilders	cucit
jc	cheaptickets	com site	circuit free	thelighterside	myhomeideas
sears	eharmony	ebay shop	macys jewelry	allgangbang	carnivalcruises

Table 1: The top 20 query-ad diagonal and off-diagonal word pairs, and the query-term absence words, most correlated with the click. Notice that some diagonal pairs appear in the off-diagonal pairs list. The query-term absence features are with respect to the top 65K off-diagonal pairs. See Figure 2 for the correlations.

as possible to the ME model. We then let the model automatically discover such relationships during training using the click-view data. For this purpose, we construct a set of binary word-pair indicator features between the query and ad terms. The query and ad are treated as bags of words for this purpose. For example, if the user query is `quality jobs` and the ad abstract contains the term `seekers`, then one possible feature for the ME model is the existence of the word pair `(jobs, seekers)`. There are a large of number of such pairs, and some form of feature selection is necessary. We use the correlation with click to eliminate the pairs with the least potential.

In this basic paradigm, we consider the following variations, similar to the variations we considered for the language-modeling features in Section 3.2.

1. **Diagonal Word Pairs**. Analogous to the identity-translation modeling, this approach only considers the pairs in which the query and ad words are identical. Again, it has the limitation that only the basic syntactic matches are captured.

2. **Off-Diagonal Word Pairs**. Analogous to the full translation modeling, this approach can accept pairs of different words in the query and ad. This flexibility allows the model to learn both positive and negative triggers between the query and ad.

3. **Query-Term Absence**. Besides the presence of word pairs, we also add a set of features that capture whether some query terms are not covered by the ad terms via the pair features (not every word pair is a feature). The binary query-term absence feature for a particular query term turns on whenever that query term does not have any associated pair in the ad. Therefore, each query term will fire some feature in ME model, which in turn can provide some form of query-length normalization, akin to length normalization in language modeling based retrieval [25]. (Notice that it would be redundant to include analogous query-term *presence* features because of the ME parameterization.)

The indicator features are chosen mainly because the ME paradigm is especially suited for large but sparsely active feature sets. With a slight abuse of notation, these features would appear in Equation (1) as

$$\sum_{i,j}\sum_{v\in V} w_v^1\ \delta_{vv}(q_i,a_j) + \sum_{i,j}\sum_{v,w\in V} w_{vw}^2\ \delta_{vw}(q_i,a_j) + \sum_{i}\sum_{v\in V} w_v^3\ \delta_v(q_i)\ \mathrm{qta}(v,\{a_j\}) \quad (5)$$

where q_i and a_j are the query and ad, respectively, terms, $\mathrm{qta}(v,\{a_j\})$ the query-term absence features, w^{1-3} the corresponding ME weights, and δ_v and δ_{vw} the univariate and bivariate, respectively, binary indicator functions. Roughly speaking, the first two terms in Equation (5) capture linear and bilinear, respectively, interactions between the query and ad with respect to a lexicon.

Table 1 shows the top diagonal and off-diagonal word pairs, and query-term absence words that are most correlated with the click; the correlation values are given in Figure 2 (see Section 4 for the details). The top pairs seem to be particularly informative for predicting click. The diagonal pairs include brand advertisers, and commerce and social sites. The off-diagonal pairs additionally include commercial terms. The query-term absence features consist of other advertisers, popular web sites, and possible typos, e.g., `njmv` and `cucit`, which could not be covered by the word pairs. The content is highly commercial, suggesting that the pairs learned in another corpora such as the web search results might not be as useful. A few pairs are superfluous, and point to the need for phrases, e.g., `(city, circuit)`, and better text normalization, e.g., `(city, $24)`.

4. EXPERIMENTS

We conducted a series of experiments comparing vector-space, language-modeling, and word-pair features for sponsored search click prediction. We also investigated the utility of different sources of text, and the effects of page presentation and click-history feedback on the new features.

4.1 Data

The training and testing data are the click-view logs collected from Yahoo! search engine traffic. Notice that the click-view logs include only the ads that were retrieved at the time, not the full candidate list. The training data is randomly selected from a continuous three-week period, and the test data from the week immediately following that period. The training and testing are non-overlapping in terms

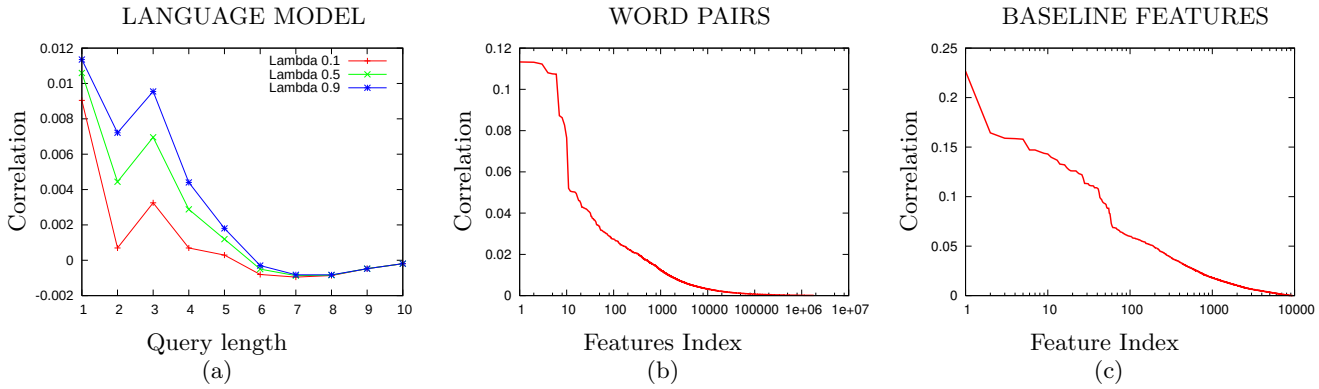

Figure 2: Correlations with click for the query-length based language modeling scores, word-pair indicator features, and baseline features. The baseline features include the click-history feedback features.

Feature	Correlation
Cosine distance, query-keyword	0.0418
Cosine distance, query-title	0.0258
Cosine distance, query-abstract	0.0020
Language model, $\lambda = 0.5$	0.0145

Table 2: Correlation with click for the cosine distance and language modeling scores.

of users as determined by the bcookie. There are about 2.8 billion training and 110 million testing examples. Each example consists of the click, query, ad, and any associated context. There are about 130 unique ads and 100 million unique queries. Some basic text normalization including the removal of punctuation and stopwords is performed for both the query and ad. A dictionary of about 100K terms consisting of the terms from the ads that have been clicked at least 50 times in the training data is compiled—the remaining terms are ignored. For the word-pair features, we use a set of about 2.75 million query-ad word pairs that had a minimum correlation of 10^{-4} with the click. As we will see in Section 4.4, this threshold is highly conservative.

4.2 Model

We use the ME model described in Section 2 to test the new features. The baseline ranking system has a number of syntactic and click-history feedback features in the context of which the new features are evaluated. The syntactic features are basic string-matching proportion features between the query and ad texts. The click-history feedback features are rank-normalized click-through rates, e.g., [6, 37], extracted at a hierarchy, starting from individual query-ad pairs. Continuous values are binned, and the conjunctions between individual features are employed as well. The resulting baseline system has about 10K parameters. As mentioned before, the click-feedback history features, if present, tend to be a strong predictor of the click. To gauge the effectiveness of the new features in a realistic setting, we test them both with and without the history features. Training and testing are done in Hadoop map-reduce framework.

4.3 Evaluation

The popular ranking metrics such as normalized discounted cumulative gain [14], or precision@n are not directly applicable in the click-view experimental paradigm without the editorial judgements. We use the precision-recall (PR) curves for evaluation, using the views with clicks as the positive class and the remaining views as the negative class. PR is sensitive to the performance on each class, which is important for sponsored search evaluation since only a small portion of the views are clicked. Instead of reducing the information in an PR curve to a summary statistic such as the area under curve, we present the full curve, comparing the performance at different precision-recall tradeoffs. In particular, the performance in the low-recall and high-precision region seems to be a better indicator of the live performance, given that few ads are displayed, and even fewer noticed.

We also use normalized cross-entropy (NCE) to measure the amount of information features X has about click C:

$$NCE \equiv \frac{H(C) - H(C|X)}{H(C)} = \frac{I(C;X)}{H(C)} \tag{6}$$

where $H(C)$ is the entropy of C, $H(C|X)$ the conditional entropy of C given X, and $I(C;X)$ the mutual information between C and X [8]. Theoretically, NCE is between 0 (null feature set) and 1 (perfect prediction). We estimate $H(C)$ and $H(C|X)$ using empirical averages on the test data, using the prior and posterior, cf. Equation (1), respectively, click probabilities.

The page placement introduces a position bias, strongly influencing whether or not the users click on ads. The eye-tracking studies have shown that a so-called golden triangle on top of the page gets the most user attention [13]. To factor out such effects, we will present results on subsets of the test data according to whether few or many ads are displayed on top (or north) of the page.[1] Similarly, we present results according to how much click history was available.

4.4 Results

To gain some insight into the new features, we first report correlation with click in Figure 2 and Table 2. We observe that even though the vector-space features fare somewhat better than the language-modeling features, neither

[1]The search engines typically display more north ads for highly commercial queries. Therefore, page-placement effects are confounded by commercialness in our data set.

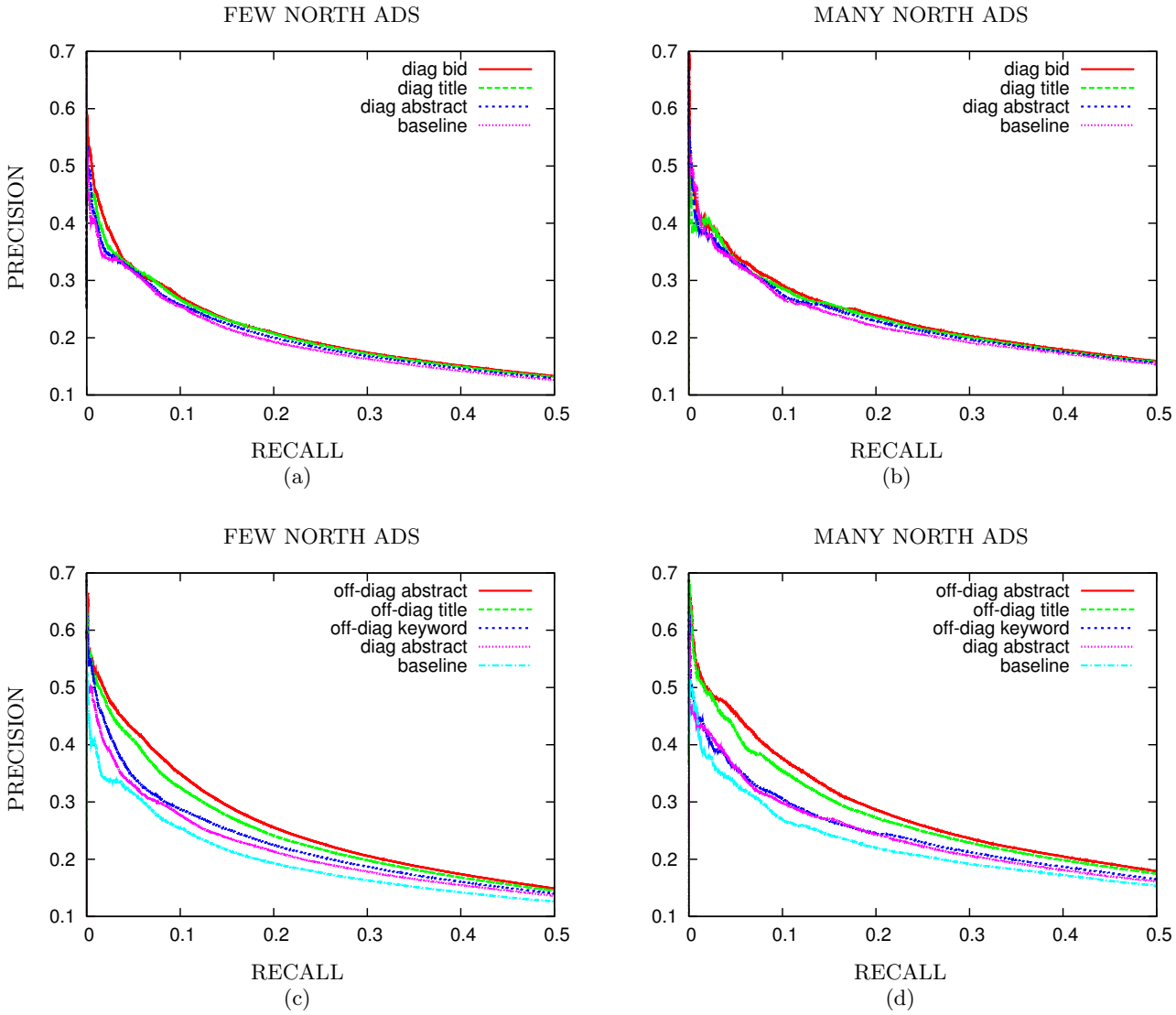

Figure 3: PR for the ME models using the diagonal and off-diagonal word-pair features are on the top and bottom, respectively, rows. PR for the slates with few and many north ads are on the left and right, respectively, columns. All models are without click-history feedback.

has much correlation with the click. On the other hand, the word-pair features exhibit strong correlation. The correlation values for the word-pair features decrease sharply (Figure 2 (b)), but there are many pairs. Also notice that the baseline features, which consist mostly of the history features, have significantly higher correlation (Figure 2 (c)). For vector-space features, the correlations values drop with the text length, probably because of the fact that bidded keywords are concise and highly informative, without the ubiquitous and filler words that are more likely to be found in the title or abstract. In addition, for language modeling, the variations such as the full-translation model and score binning did not bring in any significant improvement.

We also tested the new features in the ME model. In agreement with the correlation numbers reported above, we did not find any significant benefit from the use of either vector-space or language-modeling features, with or without the click-history feedback. On the other hand, the word-pair features proved to be highly effective. In the first set of experiments, we explored their utility without click-history feedback. When only the diagonal pairs are used (Figures 3 (a) and (b)), there is a consistent improvement, and the improvement is largest for the pairs using bidded keywords, then ad title, and finally abstract. A similar ordering was observed with the vector-space features in Table 2. Next, we added the query-term absence features (not shown), which did not significantly improve the performance of the pair features using bidded keywords, but the boosted the performances of those using title and abstract so that the performances of the all three were comparable. Next, we allowed for off-diagonal pairs which gave a significant boost in performance (Figures 3 (c) and (d)). With off-diagonal word pairs, the abstract becomes most helpful, probably due to the fact that there is more information in the abstract. In

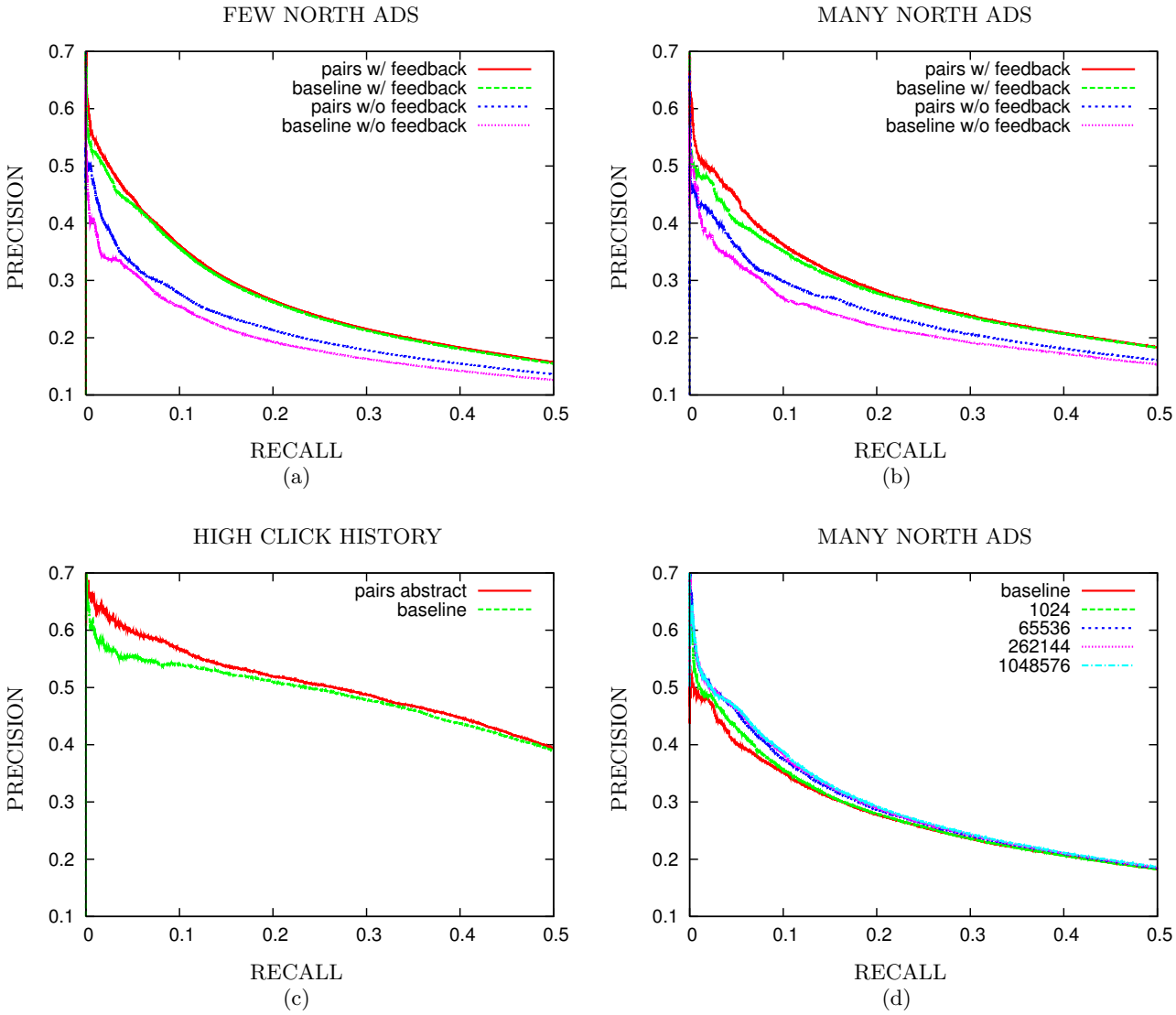

Figure 4: PR for the ME models using the off-diagonal word-pair features in combination with the click-history feedback, for the slates with few north ads (a), many north ads (b), and a lot of high click history (c). PR for varying number of word-pair features extracted from abstract are in (d).

addition, the query-term absence features no longer equalizes the performances of the pair features coming from bidded keywords, title, and abstract. Therefore, it seems that the absence features normalize mainly for the query length.

In the second set of experiments, we used the word-pair features in combination with the click-history feedback features. The feedback features are highly effective when there is sufficient history, and it is not obvious whether the word-pair features would continue to be useful in their presence. The results in Figures 4 (a), (b), and (c) indicate that they do. To some degree, the pair features compensate for the lack of history (Figures 4 (a) vs. (b)). They continue to be helpful even with high click-history feedback, and they seem to be complementary to the feedback features (Figure 4 (c)). As a final experiment, we investigated how many pair features are crucial for good performance (Figure 4 (d)). The results indicate that while the most of the gain comes from relatively few number of pairs, a large number of pairs are necessary to provide coverage in the long tail.

Comparing the performances on the views with few north ads to those with many north ads (columns one vs. two in Figures 3 and 4), we observe that while the pair features are helpful in both cases, they are more effective in the former case, possibly due to the fact those ads do not have as much click history, and there is more room for improvement. The NCEs reported in Table 3 give some insight into the contribution from the various pair features. The most contribution comes from the off-diagonal pairs, and the least from the diagonal pairs. As expected, the relative improvements are lower when the pair features are used in conjunction with click-history features. Nevertheless, the new features still contain significant predictive information about the click. All pairs of models in Table 3 are statistically different according to the asymptotic χ^2 likelihood-ratio test.

Model	NCE	
	w/o history	w/ history
Baseline	0.237	0.271
+ Diagonal pairs	0.241	0.272
+ Query-term absence	0.250	0.274
+ Off-diagonal pairs	0.266	0.277

Table 3: NCEs for the word pairs with and without click-history feedback features (higher the better).

4.5 Discussion

We can derive a number of conclusions based on the results in Section 4.4: (1) while the vector-space and language-modeling features do not bring any significant benefit in our experimental paradigm, the word-pair features do; (2) the off-diagonal pairs are the most effective word-pair features; and (3) the abstract is the most useful text source. In general, the longer the text, the more helpful the pair features. The main question is why the score-based approaches, in particular the vector-space and language modeling, were not effective when used with simple string-matching proportion features. All features had access to the same raw information. The key difference is that with the word pairs, the task of learning useful features is left up to the ranking model, which optimizes for the click prediction performance. On the other hand, in vector-space and language modeling, we are reducing the feature space for the ME model by using the term frequency-inverse document frequency and probability, respectively, scores. Those scores might be optimal for some task such as text compression, but not necessarily for click prediction.

We note that our results can be biased by the fact that we are re-ranking a retrieved list of candidates (not the full candidate list), and the retrieval might in effect capture some of the impact of the vector-space or language-modeling features. For instance, if we had a completely random set of ads, those features might have more impact. Also, the word pairs would increase dramatically with a random set of ads, and need to be filtered out. We also note that there are other more sophisticated techniques that can benefit some, or all of these approaches. For vector-space modeling, latent semantic indexing can uncover associations beyond exact match, similar to the off-diagonal pair features [9]. For language modeling, a state-of-the-art method such as Kneser-Ney smoothing might perform better [4]. Since the ads are short, smoothing is critical. We only used unigrams; longer units such as bigrams, phrases, and compound words might prove to be useful. Simply getting more text sources can also be effective. So far, we have used the bidded keywords, title, and abstract. Since the ad points to a landing page, one can use that page, or even the anchor text. Since the user does not see these sources before clicking, a comparison of the landing page with the displayed texts might be interesting for detecting spam, or other problematic ads.

5. RELATED WORK

The use of text features in IR has a long history. The vector-space model is one of the most popular models for unstructured text data [31, 30, 12]. There is a large body of work on language-modeling based IR; see, e.g., [35, 22, 21]. The translation model that we used is described in [1, 15].

While click prediction is similar to finding relevant documents, there are some important differences. The ads are short, the presentation effects are strong, and there are other concerns such as pricing which requires calibrated probability estimates. Instead of editorial judgements, sponsored search learning tends to utilize the click-view data, which is noisy, but cheap, and high volume. In addition, the dynamic features such as click-through rate can be particularly helpful [17, 6]. These and other sponsored search problems have received significant attention in recent years; see, e.g., [29, 5, 26, 34]. It is previously found out that learning from the clicks is effective, and basic semantic correlation features are more useful than the cosine similarity [5]. A comparison of the vector-space and language modeling techniques for query-to-ad matching appears in [27]. There is also related work in contextual advertising [28, 33, 24], and query rewriting [19, 36]. Ranking documents with the logistic regression model has been previously studied in [7]. The individual word features (without the pairs) have been previously used for text classification in, e.g., [16].

While our work is similar to the previous work in finding out that there are more effective text features beyond the vector-space and language modeling, it is different in a number of aspects. First, with the proposed word-pair indicator features, the feature selection is done by the ranking model in a data-driven fashion. Second, it is shown that the new features continue to be effective within a state-of-the-art system incorporating click-history feedback features. Third, the amount of the click-view data that we experiment with is one of the largest so far. Therefore, the observed differences are unlikely to be due to random effects.

6. CONCLUSIONS AND FUTURE WORK

In this paper we proposed novel word-pair indicator features for sponsored search click prediction, and showed their utility in experiments with the click-view data. Unlike vector-space and language-modeling features, the word-pair indicator features proved to be capable of learning syntactic and semantic associations between the query and ad in a data-driven manner. These detailed features give the model flexibility to sort through a large set of possibilities. Experiments with a competitive ranking system showed that the pair features significantly outperform the traditional IR features, and continue to be effective in the presence of the click-history feedback features.

Fueled by the growing online advertisement budgets, the click prediction is likely to continue to garner more attention in the future. The word-pair indicator features can provide a simple and direct method for using the query and ad texts for click prediction without restrictive assumptions. The present work can be improved upon in a number of directions. Longer units such as bigrams, and compound words in general should improve accuracy. Instead of a closed dictionary, hashing might prove to be helpful by increasing coverage at the expense of collisions [32]. Similarly, the pair features can be defined over query, or ad clusters [2]. The landing page, anchor text, or even the display URL can be used for extracting pair features. The ME model training can be modified to use an l_1-penalty function instead of the l_2-penalty function (i.e., the Gaussian prior) used in this work, to encourage sparse solutions [20].

Acknowledgments

The authors thank D. Hillard, E. Manavoğlu, and H. Raghavan of Yahoo! Inc., for discussions and feedback. This work in part was done while the first author was at Yahoo! Inc.

7. REFERENCES

[1] A. Berger and J. Lafferty. Information retrieval as statistical translation. In *Proc. ACM SIGIR*, pages 222–229, 1999.

[2] A. Broder, M. Fontoura, V. Josifovski, and L. Riedel. A semantic approach to contextual advertising. In *Proc. ACM SIGIR*, pages 559–566, 2007.

[3] D. Chakrabarti, D. Agarwal, and V. Josifovski. Contextual advertising by combining relevance with click feedback. In *Proc. WWW*, pages 417–426, 2008.

[4] S. Chen and J. Goodman. An empirical study of smoothing techniques for language modeling. *Computer Speech and Language*, 13:242–259, 1999.

[5] M. Ciaramita, V. Murdock, and V. Plachouras. Online learning from click data for sponsored search. In *Proc. WWW*, pages 227–236, 2008.

[6] C. Clarke, E. Agichtein, S. Dumais, and R. White. The influence of caption features on clickthrough patterns in web search. In *Proc. ACM SIGIR*, pages 135–142, 2007.

[7] W. Cooper, F. Gey, and D. Dabney. Probabilistic retrieval based on staged logistic regression. In *Proc. ACM SIGIR*, pages 198–210, 1992.

[8] T. Cover and J. Thomas. *Elements of Information Theory*. Wiley-Interscience, 1991.

[9] S. Deerwester, S. Dumais, G. Furnas, T. Landauer, and R. Harshman. Indexing by latent semantic analysis. *Journal of the American Society for Information Science*, 41:391–407, 1990.

[10] R. Duda, P. Hart, and D. Stork. *Pattern Classification*. Wiley-Interscience, 2000.

[11] 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*, 97:242–259, 2007.

[12] N. Fuhr. Optimum polynomial retrieval functions based on the probability ranking principle. *ACM Trans. on Information Systems*, 7(3):183–204, 1989.

[13] L. Granka, T. Joachims, and G. Gay. Eye-tracking analysis of user behavior in WWW search. In *Proc. ACM SIGIR*, pages 478–479, 2004.

[14] K. Järvelin and J. Kekäläinen. IR evaluation methods for retrieving highly relevant documents. In *Proc. ACM SIGIR*, pages 41–48, 2000.

[15] J. Jeon. *Searching Question and Answer Archives*. PhD thesis, University of Massachusetts, 2007.

[16] T. Joachims. Text categorization with support vector machines: Learning with many relevant features. In *Proc. ECML*, pages 137–142, 1998.

[17] T. Joachims. Optimizing search engines using clickthrough data. In *Proc. ACM SIGIR*, pages 133–142, 2002.

[18] T. Joachims, L. Granka, B. Pang, H. Hembrooke, and G. Gay. Accurately interpreting clickthrough data as implicit feedback. In *Proc. ACM KDD*, pages 154–161, 2005.

[19] R. Jones, B. Rey, O. Madani, and W. Greiner. Generating query substitutions. In *Proc. WWW*, pages 387–396, 2006.

[20] K. Koh, S. Kim, and S. Boyd. An interior-point method for large-scale $l1$-regularized logistic regression. *Journal of Machine Learning Research*, 8:1519–1555, 2007.

[21] O. Kurland and L. Lee. Corpus structure, language models, and ad hoc information retrieval. In *Proc. ACM SIGIR*, pages 194–201, 2004.

[22] V. Lavrenko. *A Generative Theory of Relevance*. PhD thesis, University of Massachusetts, 2004.

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

[24] V. Murdock, M. Ciaramita, and V. Plachouras. A noisy-channel approach to contextual advertising. In *Proc. KDD Workshop on Data Mining and Audience Intelligence for Advertising*, pages 21–27, 2007.

[25] J. Ponte and W. Croft. A language modeling approach to information retrieval. In *Proc. ACM SIGIR*, pages 275–281, 1998.

[26] 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 *Proc. ACM SIGIR*, pages 403–410, 2008.

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

[28] B. Ribeiro-Neto, M. Cristo, P. Golgher, and E. Moura. Impedance coupling in content-targeted advertising. In *Proc. ACM SIGIR*, pages 496–503, 2005.

[29] M. Richardson, E. Dominowska, and R. Ragno. Predicting clicks: Estimating the click-through rate for new ads. In *Proc. WWW*, pages 521–530, 2007.

[30] G. Salton and C. Buckley. Term weighting approaches in automatic text retrieval. *Information Processing and Management*, 24:513–523, 1988.

[31] G. Salton, A. Wong, and C. Yang. A vector space model for automatic indexing. *Communications of the ACM*, 18:613–620, 1975.

[32] D. Talbot and M. Osborne. Randomised language modelling for statistical machine translation. In *Proc. ACL*, pages 512–519, 2007.

[33] W. Yih, J. Goodman, and V. Carvalho. Finding advertising keywords on web pages. In *Proc. WWW*, pages 213–222, 2006.

[34] W. Yih and C. Meek. Consistent phrase relevance measures. In *Proc. KDD Workshop on Data Mining and Audience Intelligence for Advertising*, pages 37–44, 2008.

[35] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proc. ACM SIGIR*, pages 334–342, 2001.

[36] W. Zhang, X. He, B. Rey, and R. Jones. Query rewriting using active learning for sponsored search. In *Proc. ACM SIGIR*, pages 853–854, 2007.

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