Comparing Click Logs and Editorial Labels for Training Query Rewriting

Wei Vivian Zhang and Rosie Jones
Yahoo! Inc
3333 Empire Ave
Burbank, CA, 91504 USA
{zhangv,jonesr}@yahoo-inc.com

ABSTRACT

Clicks on web advertisements in response to web search queries is a major source of revenue for search companies. Query rewrites can significantly increase the coverage of web advertisements. In previous work we focused on optimizing the relevance between the query issued by the web searcher, and rewritten queries used to place advertisements. In this preliminary study, we examine some features of query rewrites which are predictive of click-throughs on sponsored search listings retrieved for those rewrites, by mining web search-click logs. We also compare the features which are predictive of relevance (judged by human editors) and the clicks in user query logs during query rewriting. Our preliminary results suggest that similar features are predictive, and so we may be able to train our models on click log data in place of human relevance judgments.

Categories and Subject Descriptors

H.3.3 [**Information Storage and Retrieval**]: Query Formulation Search Process,Retrieval Models

General Terms

Algorithms, Performance, Experimentation

Keywords

Web search query logs, click logs, query rewriting

1. INTRODUCTION

Traditionally, models used in information retrieval and web search are trained and evaluated on human relevance judgements. However, in a web search setting, we have implicit relevance judgements from web searchers based on their click behavior [3]. In a laboratory or professional industrial setting, human relevance judgements may be made in a thoughtful way. On the other hand, clicks are generally made very quickly. Thus a machine-learned model trained on relevance judgments may differ from a model trained on click data. This is particularly true in a setting such as query rewriting, where editorial judgments may find synonym substitutions acceptable, while user clicks may imply a preference for rewrites which are similar in appearance, for example very small changes in wording and spelling. In this work we compare the features of rewrites which are highly predictive of editorial relevance judgments, to those which are predictive of clicks.

In Section 2 we give an overview of a system for automatically rewriting queries for sponsored search, and describe how it is trained using human relevance judgements. In Section 3 we look at the clickthrough rates for different editorially-rated relevance scores and see that there is a relationship: more relevant rewrites tend to receive more clicks. In Section 4 we look at ways of using clicks logs as training data for learning relevance functions. In Section 5, we look at individual features considered and how click-rate varies as we vary the feature values. We then look at coefficients learned when we use click data to train the same set of features as trained on human judgements, and see that similar coefficients are learned.

WWW2007, May 8–12, 2007, Banff, Canada.

2. AUTOMATIC QUERY REWRITING FOR SPONSORED SEARCH

Given a web search query, we assume a method for presenting relevant advertisements to the user. However, a query is not always a perfect description of the user's information need. For example, if the user query has a spelling error, we may do better spell-correcting the query before looking for relevant advertisements. More generally, we may substitute synonyms or perform other modifications to the query, to expand the range of possible advertisements which can be retrieved. Jones et al [4] describe a system for automatically generating rewritings for queries.

The approach to generate query rewrites is twofold. First, sequential queries from web searchers are mined as a source of related queries and terms. Then a machine-learned model is trained to identify the most related rewrites and score them by predicting the editorial relevance judgment. The training is based on manual relevance judgements, using the scheme given in Table 1. After evaluating a large number of features, Jones et al [4] fit a linear model using just three features, given in Equation 1:

$$\begin{aligned} f(q1,q2) = \quad & 0.74 \quad +1.88\ \text{editDist}(q1,q2) \\ & \quad\ \ +0.71\ \text{wordDist}(q1,q2) \\ & \quad\ \ +0.36\ \text{numSubst}(q1,q2) \end{aligned} \tag{1}$$

where editDist is the character edit distance, wordDist is a word overlap feature described in more detail in Section 5.4, and numSubst is the number of phrases changed between

Score		Definition	Example
1	Precise Match	A near-certain match.	corvette car - chevrolet corvette
2	Approximate Match	A probable, but inexact match with user intent.	apple music player - ipod shuffle
3	Marginal Match	A distant, but plausible match to a related topic.	glasses - contact lenses
4	Mismatch	A clear mismatch.	time magazine – time and date magazine

Table 1: Editorial Scoring System for Query Rewrites

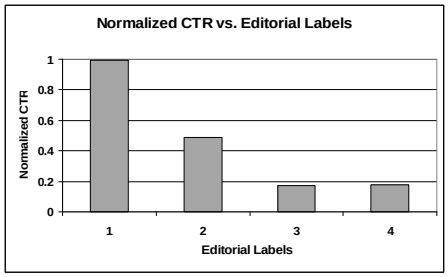

Figure 1: Relevance judgements from human editorial labeling of query rewrites are validated with their click-through rates on sponsored search results.

the original query and rewritten query. In this model, the higher the predicted score, the worse the relevance.

3. COMPARISON BETWEEN EDITORIAL LABELS AND CLICK THROUGH RATE

Jones et al [4] evaluated their query rewrite system on the basis of a four-point scale of editorial relevance judgments. However, they did not provide any information about how these relevance judgements relate to likely clickthrough on sponsored search results. In order to quantify the relationship between their relevance judgments and clickthrough rates on sponsored search results, generated using query rewrites, we randomly sampled 2000 queries from a web search log. For each of these queries, up to five rewrites are generated using automatic query rewriting [4]. A professional editorial team hand-labeled each pair (original query, rewritten query) using the scoring system given in Table 1. During initial training the ditors compared their scores to calibrate their labeling. We then identified occurrences of the query-rewrite pair in a search-click log and calculated the clickthrough rate for each class of query-rewrite. In Figure 1 we see that pairs with relevance score of 1 (precise match) have the best clickthrough rate, and that clickthrough rate is lower for the lower relevance classes. Queries with better editorial relevance do indeed see higher clickthrough rates.

Some rewritten queries have low relevance with respect to the original queries, but still receive user clicks. Table 2 gives sample pairs labeled as marginal match and mismatch which received clicks. This suggests that click data is noisy: even irrelevant results receive some clicks. When we train machine-learned models with click information as labels, we will need to be careful that our methods are robust to noise.

4. CLICK LOGS AS TRAINING DATA

Dupret et al [2], give a theoretical model for the rank-position effects of click-through rate, as well as empirically estimating the rank-position effect from data. They also

Marginal-match Rewrites	
Original query	**Rewritten query**
free software downloads	free downloads
wcw	ecw
sidekick 3	sidekick 2
superpages	white pages
centennial wireless	cingular wireless
helicopter game	game
usda	fda
superpages	yellow pages
sidekick 3	sidekick
airjamaica.com	aa.com
spanish translator	spanish translation
usps	ups
craigslist.com	monster.com
white pages	yellow pages
Mismatch Rewrites	
Original query	**Rewritten query**
nitric acid	nitric oxide
contender	contender boats
pool	pool tables
girl	cheetah girl
trec	trek
monkeys	sea monkeys
rds	rs
u	us
fights	flights
thong pics	thong
white pages	zip codes

Table 2: Even query rewrites manually identified as marginal matches and mismatches received some clicks.

build theoretical models for modeling search engine quality. Joachims et al [3] use click data to infer a ranking for documents by inferring relative relevance judgments. For example, a click at rank i means the result is more relevant than the result shown at rank $i-1$. In principal, one could use that ranking to train a relevance model. However, it is useful only when each instance has been seen enough times to learn a ranking. In practice, search engine queries follow a Zipf distribution, so many queries will not have been seen frequently enough to learn a ranking for the documents shown in response to that query. Instead, we may wish to use information even from rare events to train our model. Regelson and Fain [5] show that for rare queries, we can use information such as related queries to estimate click-through rates. We will be considering features derived from query pairs as the way to aggregate over rare events.

4.1 Clicks Normalized By Expected Clicks

There is a *rank effect* in web search; users tend to click on results ranked higher, regardless of the quality of those results [3]. Agichtein et al [1] normalize click data by subtracting the number of clicks expected for a result at a given rank, from the number of clicks actually seen at a given rank. We use a similar approach, but instead of normalizing for expected clicks with subtraction, we use division (which is equivalent under a log likelihood model). Let clickthrough rate be $ctr(r) = \frac{c_r}{i_r}$ where r is the rank, c_r is the total number of clicks on results at rank r, and i_r is the total number of impressions of results at rank r. Then for a result or set of results m we can talk about their expected clicks at

a certain rank: $ec(r,m) = ctr(r)i_r(m)$. The total number of clicks expected for m at all ranks is $\sum_r ec(r,m)$ and we define *clicks over expected clicks* (COEC) as

$$\frac{\sum_r c(r,m)}{\sum_r ec(r,m)} \tag{2}$$

where c(r,m) is the number of actual clicks on result (set) m at rank r. We can then use COEC as our target function for machine learning: it has the advantage of being rank-normalized.

4.2 Binary Clicks as Training Data

We can use click data to train a relevance model if we perform logistic regression, and consider rank as one of the features in the model. Our task is then to predict $p(click) = p$. The general form of the model is

$$\log \frac{p}{(1-p)} = c + \sum_i w_i f_i \tag{3}$$

where f_i are our features which may correlate with relevance or clicks, and w_i are the weights we wish to learn.

4.3 Data for Training from Click Logs

In order to train from click logs we need to store information about historical searches and clicks. In particular we need: <query string, results> where for the each of the results results we need <rank, clicked-or-not>. We do not need session information or any kind of user identifier. In order to train with this kind of data we need hundreds of millions of queries paired with result clicks.

The rewrites which are input to the system are trained on hundreds of millions of sequential query pairs: we need to store the query string and a temporary anonymized identifier allowing us to identify pairs of sequential queries from the same anonymous user.

5. FEATURES CORRELATED WITH CLICKS

Our goal is to generate high quality query rewrites. In order to use click logs as our source of training data for machine learning, we need to identify features of query rewrites which correlate with increases in COEC. In this section we look at the correlation of these and other features with COEC on sponsored search results for rewritten queries. Except ranks, the features are all extracted from pairs of <original query, rewritten query>, where rewritten queries differ in at least one character from the original query.

5.1 Rank

Rank is one of the dominant features explaining the distribution of clicks on search results [3], [5]. Figure 2 shows the click-through rate at different ranks. The click-through rate decreases dramatically when rank decreases, while increasing at rank 10, which usually corresponds to the results shown at the bottom of the web page. Ranks 1, 2, 3 and 4 have the highest click-though rates. This confirms the need for rank-normalization or rank0modeling on our dataset.

5.2 Length Difference

We observed that when the original and rewritten queries have similar length in characters, the resulting advertisement is more likely to be clicked on. Figure 3 shows how

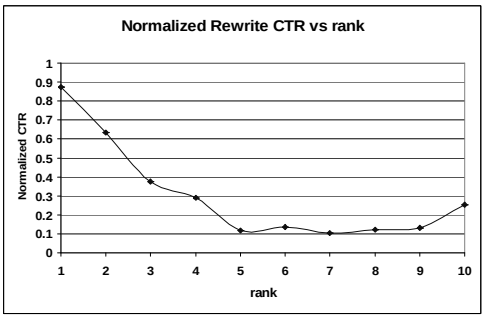

Figure 2: Normalized click-through rate on sponsored search for rewritten queries, versus the advertisement display rank.

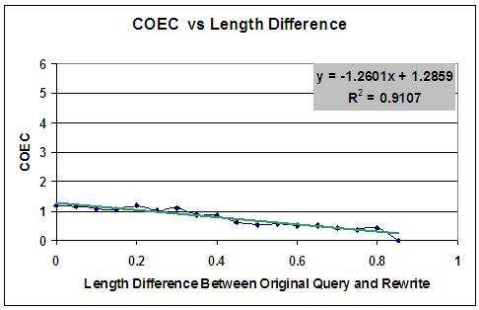

Figure 3: When we rank normalize using clicks-over-expected-clicks (COEC) the difference in length between the query and its rewrite appears to be predictive of clicks, with a correlation of 0.91.

COEC varies as the length difference between the query and its rewrite varies. COEC aggregates information from all ranks, and has a strong correlation with query length difference, with $R^2 = 0.91$.

5.3 Edit Distance

In Figure 4 we see COEC as we vary the normalized character edit distance between the query and the rewrite. Edit distance is anti-correlated with clickthrough rates for rewrites. The smaller the edit distance, the higher the click-through rate. This is likely because small-edit distance rewrites tend to be spelling corrections and different morphological forms of terms.

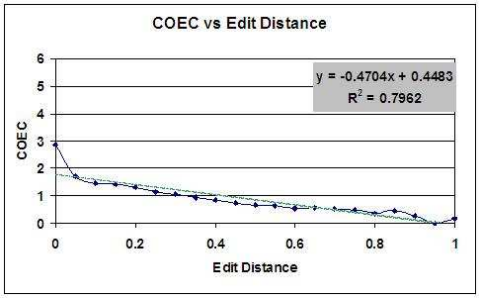

Figure 4: COEC versus character edit distance for query rewrites. We see that edit distance is correlated with COEC at 0.796.

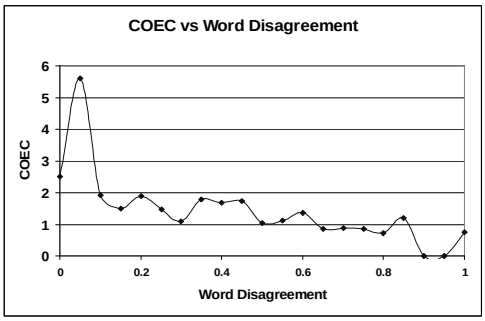

Figure 5: COEC versus word disagreement

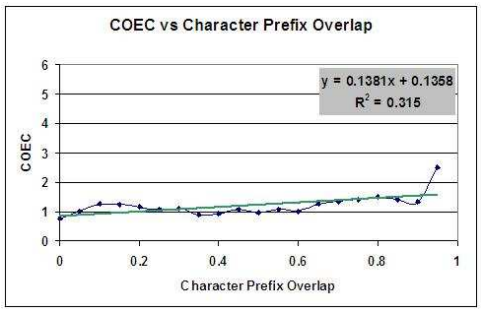

Figure 6: Character prefix overlap is predictive of clicks-over-expected-clicks with a correlation of 0.3.

5.4 Word Disagreement

We might expect rewrites which change few words to lead to semantically similar queries, with consequently high click-rate. The word disagreement feature is calculated as follows,

$$wordDist = 1 - \frac{W_{q_1} \cap W_{q_2}}{W_{q_1} \cup W_{q_2}} \quad (4)$$

where W_{q_1} is the set of words in the original query and W_{q_2} is the set of words in the rewritten query.

Figure 5 shows COEC for the word disagreement feature. The smaller the word disagreement, the better. The relationship is not strictly linear, since different length queries can have different fractional numbers of words in common.

5.5 Character Prefix Overlap

Figure 6 shows COEC for different degrees of character prefix overlap. We observe that larger prefix overlap may contribute to higher clicks. This may correspond to rewrites which are stemming or morphological variants, which differ only at the end of the rewrite query words. The correlation is 0.32, so it is not as correlated as edit distance. Spelling changes and morphological changes will all be captured under the edit distance metric.

6. REWRITE MODEL TRAINED ON CLICKS

[4] reported the predictive model for query expansion relevance prediction given in Equation 1. We used the same three features and learned a preliminary model using logistic regression:

$$\begin{aligned} \log \frac{p}{(1-p)} = \quad & -3.94619 \quad -1.08 \text{ editDist}(q1, q2) \\ & -0.57 \text{ wordDist}(q1, q2) \\ & -0.08 \text{ numSubst}(q1, q2) \quad (5) \end{aligned}$$

where p is the probability of click.

Interestingly, the relative magnitude of the coefficients for editDist and wordDist are the same: editDist is about twice the coefficient of wordDist. *numSubst* has a smaller magnitude coefficient compared with the previous model trained using human editorial labels. Thus training on click data does not lead to identical models, but can lead to similar orders of magnitude of coefficients for some features. One limitation of this experiment is that our sample data is restricted to the data generated using equation 1. An ideal setting for training from click data would use a random sample of rewrites.

7. CONCLUSIONS

Optimizing the relevance between user queries and their rewrites may not lead to optimal clicks, although generally more relevant rewrites get more clicks. We used query log analysis to show the correlation between clicks and the features generated from query rewrites. The features we study are simple syntactic features, including edit distance,word disagreement ratio, and the length difference between a query and its rewrite. There may be other features which are more predictive of rewrite quality and rewrites likely to lead to clicks on sponsored search results. We also propose a simple model trained on click data for evaluating the quality of rewrites. In future work we would like to see how models trained on relevance and click data relate to each other. For example, if we train on click data, how do the examples generated perform when evaluated editorially? Similarly, is a click-trained model better than a relevance-trained model when our end goal is to generate relevant results which lead to clicks, and by how much does the click-through rate differ under these two settings?

8. REFERENCES

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

[2] G. Dupret, B. Piwowarski, C. Hurtado, and M. Mendoza. A statistical model of query log generation. In *SPIRE*, LNCS 4209, pages 217–228. Springer, 2006.

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

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

[5] M. Regelson and D. C. Fain. Predicting clickthrough rate using keyword clusters. In *Proceedings of Second Workshop on Sponsored Search Auctions, at the ACM Conference on Electronic Commerce (EC'06)*, 2006.