Active Evaluation of Ranking Functions based on Graded Relevance (Extended Abstract)*

Christoph Sawade
University of Potsdam
Potsdam, Germany
sawade@cs.uni-potsdam.de

Steffen Bickel
Nokia gate5 GmbH
Berlin, Germany
steffen.bickel@nokia.com

Timo von Oertzen
University of Virginia
Charlottesville, VA, USA
timo@virgina.edu

Tobias Scheffer
University of Potsdam
Potsdam, Germany
scheffer@cs.uni-potsdam.de

Niels Landwehr
University of Potsdam
Potsdam, Germany
landwehr@cs.uni-potsdam.de

Abstract

Evaluating the quality of ranking functions is a core task in web search and other information retrieval domains. Because query distributions and item relevance change over time, ranking models often cannot be evaluated accurately on held-out training data. Instead, considerable effort is spent on manually labeling the relevance of query results for test queries in order to track ranking performance. We address the problem of estimating ranking performance as accurately as possible on a fixed labeling budget. Estimates are based on a set of most informative test queries selected by an active sampling distribution. Query labeling costs depend on the number of result items and item-specific attributes such as document length. We derive cost-optimal sampling distributions for commonly used ranking performance measures. Experiments on web search engine data illustrate significant reductions in labeling costs.

1 Introduction

This paper addresses the problem of estimating the performance of a given ranking function in terms of graded relevance measures such as Discounted Cumulative Gain [Järvelin and Kekäläinen, 2002] and Expected Reciprocal Rank [Chapelle *et al.*, 2009]. In informational retrieval domains, ranking models often cannot be evaluated on held-out training data. For example, older training data might not represent the distribution of queries the model is currently exposed to, or ranking models might be procured from a third party that does not provide any training data.

In practice, ranking performance is estimated by applying a given ranking model to a representative set of test queries and manually assessing the relevance of all retrieved items for each query. We study the problem of estimating the performance of ranking models as accurately as possible on a fixed budget for labeling item relevance, or, equivalently, minimizing labeling costs for a given level of estimation accuracy. Specifically, we focus on the problem of estimating performance differences between two ranking models; this is required, for instance, to evaluate the result of an index update.

We assume that drawing unlabeled data $x \sim p(x)$ from the distribution of queries that the model is exposed to is inexpensive, whereas obtaining relevance labels is costly. The standard approach to estimating ranking performance is to draw a sample of test queries from $p(x)$, obtain relevance labels, and compute the empirical performance. However, recent results on *active risk estimation* [Sawade *et al.*, 2010] and *active comparison* [Sawade *et al.*, 2012b] indicate that estimation accuracy can be improved by drawing test examples from an appropriately engineered instrumental distribution $q(x)$ rather than $p(x)$, and correcting for the discrepancy between p and q by importance weighting.

In this paper, we study active estimates of ranking performance. Section 2 details the problem setting. Section 3 derives cost-optimal sampling distributions. Section 4 presents empirical results, Section 5 concludes with a discussion of related work.

2 Problem Setting

Let $\mathcal{X}$ denote a space of queries and $\mathcal{Z}$ denote a finite space of items. We study ranking functions

$$\mathbf{r} : x \mapsto \left(r_1(x), \ldots, r_{|\mathbf{r}(x)|}(x)\right)^{\mathsf{T}}$$

that, given a query $x \in \mathcal{X}$, return a list of $|\mathbf{r}(x)|$ items $r_i(x) \in \mathcal{Z}$ ordered by relevance. The number of items in a ranking $\mathbf{r}(x)$ can vary depending on the query and application domain from thousands (web search) to ten or fewer (mobile applications that have to present results on a small screen). Ranking performance of $\mathbf{r}$ is defined in terms of graded relevance labels $y_z \in \mathcal{Y}$ that represent the relevance of an item $z \in \mathcal{Z}$

*The paper on which this extended abstract is based was the recipient of the best paper award of the 2012 European Conference on Machine Learning [Sawade et al., 2012a].

for the query x, where $\mathcal{Y} \subset \mathbb{R}$ is a finite space of relevance labels with minimum zero (irrelevant) and maximum y_{max} (perfectly relevant). We summarize the graded relevance of all $z \in \mathcal{Z}$ in a label vector $\mathbf{y} \in \mathcal{Y}^{\mathcal{Z}}$ with components y_z for $z \in \mathcal{Z}$.

The performance of a ranking $\mathbf{r}(x)$ given a label vector $\mathbf{y}$ is scored by a ranking measure $L(\mathbf{r}(x), \mathbf{y})$. Intuitively, $L(\mathbf{r}(x), \mathbf{y})$ will be high if many relevant items are ranked highly in $\mathbf{r}(x)$. In the longer version of this paper [Sawade *et al.*, 2012a], we study active performance estimation in detail for two commonly used ranking measures: *Discounted Cumulative Gain* (DCG), introduced by Järvelin and Kekäläinen [2002], and *Expected Reciprocal Rank* (ERR), introduced by Chapelle et. al [2009].

In this paper, we will mainly discuss the ERR ranking performance measure. ERR is based on a probabilistic user model: the user scans a list of documents in the order defined by $\mathbf{r}(x)$ and chooses the first document that appears sufficiently relevant; the likelihood of choosing a document z is a function of its graded relevance score y_z. If s denotes the position of the chosen document in $\mathbf{r}(x)$, then ERR is the expectation of the reciprocal rank $1/s$ under the probabilistic user model. More formally, ERR is given by

$$L(\mathbf{r}(x), \mathbf{y}) = \sum_{i=1}^{|\mathbf{r}(x)|} \frac{1}{i} \ell_{err}\left(y_{r_i(x)}\right) \prod_{l=1}^{i-1} (1 - \ell_{err}\left(y_{r_l(x)}\right))$$

$$\ell_{err}(y) = \frac{2^y - 1}{2^{y_{max}}}.$$

Both DCG and ERR discount relevance with ranking position, ranking quality is thus most strongly influenced by documents that are ranked highly. If $\mathbf{r}(x)$ includes many items, DCG and ERR are in practice often approximated by only labeling items up to a certain position in the ranking or a certain relevance threshold and ignoring the contribution of lower-ranked items.

The overall ranking performance of the function $\mathbf{r}$ with respect to the distribution $p(x, \mathbf{y})$ is given by

$$R[\mathbf{r}] = \int \sum_{\mathbf{y} \in \mathcal{Y}^{\mathcal{Z}}} L\left(\mathbf{r}(x), \mathbf{y}\right) p(x, \mathbf{y}) \mathrm{d}x. \tag{1}$$

If the context is clear, we refer to $R[\mathbf{r}]$ simply by R. Since $p(x, \mathbf{y})$ is unknown, ranking performance is typically approximated by an empirical average

$$\hat{R}_n[\mathbf{r}] = \frac{1}{n} \sum_{j=1}^{n} L\left(\mathbf{r}(x_j), \mathbf{y}_j\right), \tag{2}$$

where a set of test queries $x_1, ..., x_n$ and relevance label vectors $\mathbf{y}_1, ..., \mathbf{y}_n$ are drawn *iid* from $p(x, \mathbf{y})$.

Test queries x_i need not necessarily be drawn according to the input distribution p. When instances are drawn according to an instrumental distribution q, an estimator can be defined as

$$\hat{R}_{n,q}[\mathbf{r}] = \frac{1}{W} \sum_{j=1}^{n} \frac{p(x_j)}{q(x_j)} L(\mathbf{r}(x_j), \mathbf{y}_j), \tag{3}$$

where $(x_j, \mathbf{y}_j)$ are drawn from $q(x)p(\mathbf{y}|x)$ and $W = \sum_{j=1}^{n} \frac{p(x_j)}{q(x_j)}$. Weighting factors $\frac{p(x_j)}{q(x_j)}$ correct for the discrepancy between p and q, and ensure that the estimator is consistent (that is, $\hat{R}_{n,q}$ converges to R with $n \to \infty$). For certain choices of the sampling distribution q, $\hat{R}_{n,q}$ may be a more label-efficient estimator of the true performance R than $\hat{R}_n$ [Sawade *et al.*, 2010].

A crucial feature of ranking domains is that labeling costs for queries $x \in \mathcal{X}$ vary with the number of items $|\mathbf{r}(x)|$ returned and item-specific features such as the length of a document whose relevance has to be determined. We denote labeling costs for a query x by $\lambda(x)$. For the rest of the paper, we assume that we are given two ranking functions $\mathbf{r}_1$ and $\mathbf{r}_2$, and the goal is to find an accurate estimate $\hat{\Delta}_{n,q} = \hat{R}_{n,q}[\mathbf{r}_1] - \hat{R}_{n,q}[\mathbf{r}_2]$ of their true performance difference $\Delta = R[\mathbf{r}_1] - R[\mathbf{r}_2]$. We have additionally studied the related problem of actively estimating the absolute performance $R[\mathbf{r}]$ of a single ranking function $\mathbf{r}$ [Sawade *et al.*, 2012a].

More formally, in this paper we want to find the sampling distribution q^* minimizing the expected deviation of $\hat{\Delta}_{n,q}$ from Δ under the constraint that the expected overall labeling costs stay below a budget $\Lambda \in \mathbb{R}$:

$$q^* = \arg\min_q \left(\min_n \mathbb{E}\left[\left(\hat{\Delta}_{n,q} - \Delta \right)^2 \right] \right),$$
$$\text{s.t. } \mathbb{E}\left[\sum_{j=1}^{n} \lambda(x_j) \right] \leq \Lambda. \tag{4}$$

Note that Optimization 4 represents a trade-off between labeling costs and informativeness of a test query: optimization over n implies that many inexpensive or few expensive queries could be chosen. When applying the method, we will sample and label test queries from the distribution solving Optimization 4; the number of instances sampled in practice is determined by the labeling budget.

3 Optimal Sampling Distribution

We call a sampling distribution asymptotically optimal if it holds that any other sampling distribution produces a higher estimation error $\mathbb{E}[(\hat{\Delta}_{n,q} - \Delta)^2]$ if Λ is made sufficiently large. Theorem 1 states the asymptotically optimal sampling distribution, thereby asymptotically solving Optimization Problem 4.

Theorem 1 (Optimal Sampling Distribution) *Let* $\delta(x, \mathbf{y}) = L(\mathbf{r}_1(x), \mathbf{y}) - L(\mathbf{r}_2(x), y)$. *The asymptotically optimal sampling distribution is*

$$q^*(x) \propto \frac{p(x)}{\sqrt{\lambda(x)}} \sqrt{\sum_{\mathbf{y} \in \mathcal{Y}^{\mathcal{Z}}} \left(\delta(x, \mathbf{y}) - \Delta\right)^2 p(\mathbf{y}|x)}. \tag{5}$$

Unfortunately, the sampling distribution prescribed by Theorem 1 cannot directly be evaluated. It depends on the unknown test distribution $p(x)$, the unknown conditional distribution $p(\mathbf{y}|x)$, and the true performance difference Δ (that can be computed from $p(x)$ and $p(\mathbf{y}|x)$, see Equation 1).

Algorithm 1 Active Estimation of Ranking Performance

input Ranking functions $\mathbf{r}_1$,$\mathbf{r}_2$, graded relevance model $p(y_z|x,z;\theta)$; pool D, labeling budget Λ.

1: Compute empirical sampling distribution q^*.
2: Initialize $n \leftarrow 0$.
3: Draw $x_1 \sim q^*(x)$ from D with replacement.
4: **while** $\sum_{j=1}^{n+1} \lambda(x_j) \leq \Lambda$ **do**
5: Obtain $\mathbf{y}_{n+1} \sim p(\mathbf{y}|x_{n+1})$ from human labeler.
6: Update number of instances $n \leftarrow n+1$.
7: Draw $x_{n+1} \sim q^*(x)$ from D w/ replacement.
8: **end while**
9: Compute $\hat{\Delta}_{n,q} = \hat{R}_{n,q}[\mathbf{r}_1] - \hat{R}_{n,q}[\mathbf{r}_2]$ (cf. Equation 3)

output $\hat{\Delta}_{n,q}$.

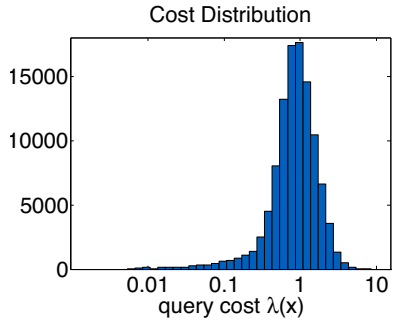

Figure 2: Distribution of query labeling costs $\lambda(x)$ in the MSLR-WEB30k data set.

To apply the method, we first move to a setting in which a pool D of m unlabeled queries is available. Queries from this pool can be sampled and then labeled at a cost. Drawing queries from the pool replaces generating them under the test distribution; that is, $p(x) = \frac{1}{m}$ for all $x \in D$. Second, we assume independence of individual relevance labels given a query x, that is, $p(\mathbf{y}|x) = \prod_{z \in \mathcal{Z}} p(y_z|x,z)$, and approximate the remaining conditional $p(y_z|x,z)$ by a model $p(y_z|x,z;\theta)$ of graded relevance. For the large class of pointwise ranking methods – that is, methods that produce a ranking by predicting graded relevance scores for query-document pairs and then sorting documents according to their score – such a model can typically be derived from the graded relevance predictors to be evaluated. Note that while these approximations might degrade the quality of the sampling distribution and thus affect the efficiency of the estimation procedure, the weighting factors in Equation 3 ensure that the performance estimate stays consistent.

Plugging these approximations into Equation 5 yields an empirical sampling distribution that can be computed on the pool of test instances. Despite the simplifying assumptions made, computing this empirical distribution is nontrivial because of the summation over the (exponentially large) space of relevance label vectors $\mathbf{y}$ in Equation 5. In the longer version of this paper, we derive polynomial-time solutions using dynamic programming for the ranking measures DCG and ERR [Sawade *et al.*, 2012a].

Algorithm 1 summarizes the active estimation algorithm. It samples queries $x_1, ..., x_n$ with replacement from the pool D according to the distribution q^* and obtains relevance labels from a human labeler for all items included in $\mathbf{r}_1(x_i) \cup \mathbf{r}_2(x_i)$ until the labeling budget Λ is exhausted. Note that queries can be drawn more than once; in the special case that the labeling process is deterministic, recurring labels can be looked up rather than be queried from the deterministic labeling oracle repeatedly. Hence, the actual labeling costs may stay below $\sum_{j=1}^{n} \lambda(x_j)$. In this case, the loop is continued until the labeling budget Λ is exhausted.

4 Empirical Studies

We compare active estimation of ranking performance (Algorithm 1, labeled *active*) to estimation based on a test sample drawn uniformly from the pool (Equation 2, labeled *passive*). Algorithm 1 requires a model $p(y_z|x,z;\theta)$ of graded relevance in order to compute the sampling distribution q^* as explained in Section 3. If no such model is available, a uniform distribution $p(y_z|x,z;\theta) = \frac{1}{|\mathcal{Y}|}$ can be used instead (labeled *active*$_{uniD}$). To quantify the effect of modeling costs, we also study a variant of Algorithm 1 that assumes uniform costs $\lambda(x) = 1$ in Equation 5 (labeled *active*$_{uniC}$).

It contains 31,531 queries, and a set of documents for each query whose relevance for the query has been determined by human labelers in the process of developing the Bing search engine. The resulting 3,771,125 query-document pairs are represented by 136 features widely used in the information retrieval community (such as query term statistics, page rank, and click counts). Relevance labels take values from 0 (irrelevant) to 4 (perfectly relevant).

We train different types of ranking functions on part of the data, and use the remaining data as the pool D from which queries can be drawn and labeled until the labeling budget Λ is exhausted. To quantify the human effort realistically, we model the labeling costs $\lambda(x)$ for a query x as proportional to a sum of costs incurred for labeling individual documents $z \in \mathbf{r}(x)$; labeling costs for a single document z are assumed to be logarithmic in the document length. The cost unit is chosen such that average labeling costs for a query are one. Figure 2 shows the distribution of labeling costs $\lambda(x)$. All results are averaged over five folds and 5,000 repetitions of the evaluation process.

Based on the outcome of the 2010 Yahoo ranking challenge [Mohan *et al.*, 2011; Chapelle and Chang, 2011], we study pointwise ranking approaches. The ranking function is obtained by returning all documents associated with a query sorted according to their predicted graded relevance. To train graded relevance models on query-document pairs, we employ Random Forest regression [Breiman, 2001], the ordinal classification extension to Random Forests [Li *et al.*, 2007; Mohan *et al.*, 2011], a MAP version of Ordered Logit [McCullagh, 1980], and a linear Ranking SVM [Herbrich *et al.*, 2000]. For the Random Forest model, we apply the approach from [Li *et al.*, 2007; Mohan *et al.*, 2011] to obtain the probability estimates $p(y_z|x,z;\theta)$ required for active estimation.

Figure 1 (top row) shows *model selection error* – defined as the fraction of experiments in which an evaluation method

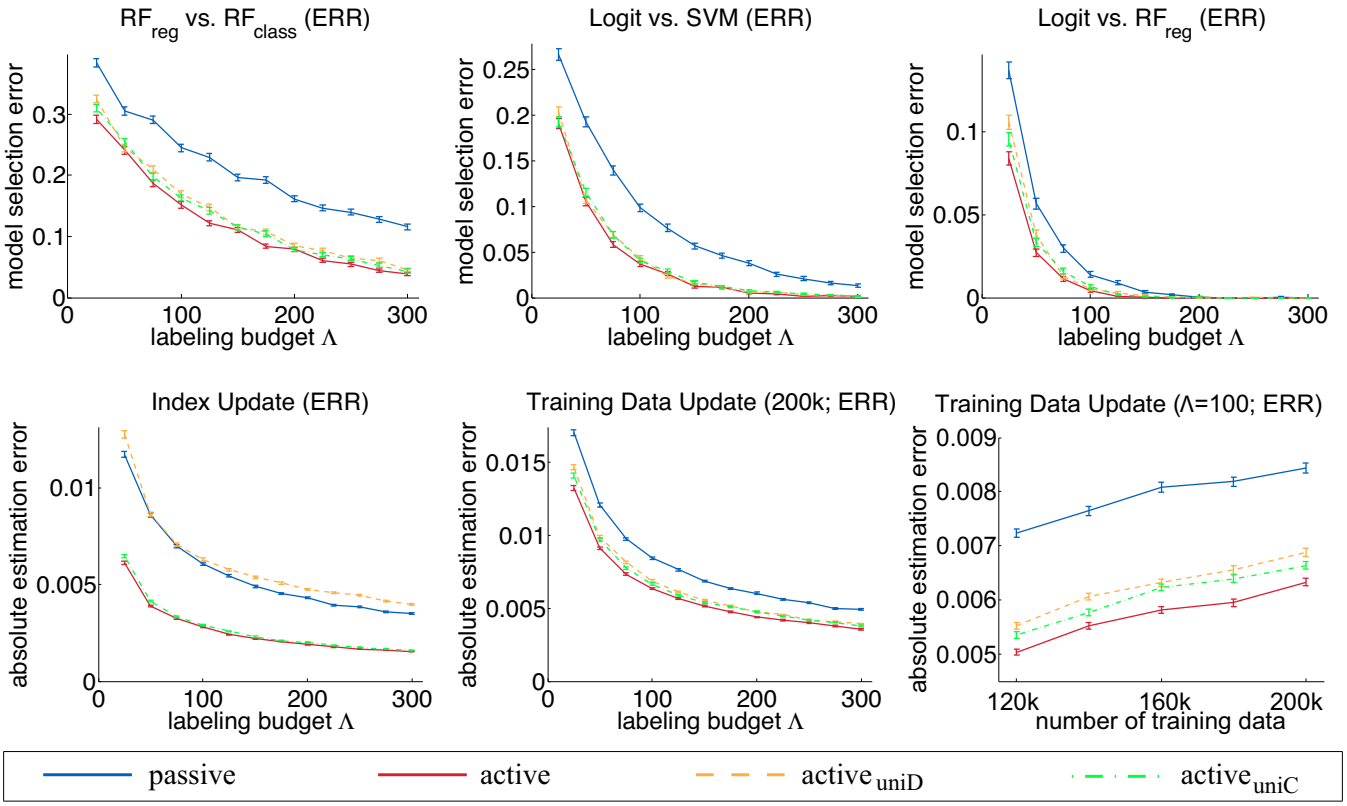

Figure 1: Top row: Model selection error over Λ when comparing Random Forest regression vs. classification (left), and Ordered Logit vs. Ranking SVM (center) or Random Forest regression (right). Bottom row: Absolute estimation error over Λ for a simulated index update (left). Absolute estimation error comparing ranking functions trained on 100,000 vs. 200,000 query-document pairs over Λ (center), and over training set size of second model at $\Lambda = 100$ (right). Error bars indicate the standard error.

does not correctly identify the model with higher true performance – when comparing different ranking functions based on Random Forest, Logit, and SVM models. Active estimation more reliably identifies the model with higher ranking performance, saving between 30% and 55% of labeling effort compared to passive estimation at $\Lambda = 200$.

As a further comparative evaluation we simulate an index update. An outdated index with lower coverage is simulated by randomly removing 10% of all query-document pairs from each result list $\mathbf{r}(x)$ for all queries; we estimate the difference in performance between models based on the outdated and current index. Figure 1 (bottom row, left) shows absolute deviation of estimated from true performance difference over labeling budget Λ. We observe that active estimation quantifies the impact of the index update more accurately than passive estimation, saving approximately 75% of labeling effort. We finally simulate the incorporation of novel sources of training data by comparing a Random Forest model trained on 100,000 query-document pairs ($\mathbf{r}_1$) to a Random Forest model trained on between 120,000 and 200,000 query-document pairs ($\mathbf{r}_2$). Figure 1 (bottom row; center and right) shows absolute deviation of estimated from true performance difference as a function of Λ and as a function of the number of query-document pairs the model $\mathbf{r}_2$ is trained on.

5 Related Work

There has been significant interest in learning ranking functions from data in order to improve the relevance of search results [Burges, 2010; Li *et al.*, 2007; Mohan *et al.*, 2011; Zheng *et al.*, 2007].

To reduce the amount of training data that needs to be relevance-labeled, several approaches for active learning of ranking functions have been proposed [Long *et al.*, 2010; Radlinski and Joachims, 2007]. The active performance estimation problem discussed in this paper can be seen as a dual problem of active learning, where the goal is to obtain accurate performance estimates rather than accurate models.

As an alternative to selecting most relevant queries, approaches that select the most relevant documents to label for a single query have also been studied. Carterette et al. [2006] use document sampling to decide which of two ranking functions achieves higher *precision at k*. Aslam et al. [2006] use document sampling to obtain unbiased estimates of mean average precision and mean R-precision. Carterette and Smucker [2007] study statistical significance testing from reduced document sets.

References

[Aslam *et al.*, 2006] J. Aslam, V. Pavlu, and E. Yilmaz. A statistical method for system evaluation using incomplete judgments. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development on Information Retrieval*, 2006.

[Breiman, 2001] L. Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[Burges, 2010] C. Burges. RankNet to LambdaRank to LambdaMART: An overview. Technical Report MSR-TR-2010-82, Microsoft Research, 2010.

[Carterette and Smucker, 2007] B. Carterette and M. Smucker. Hypothesis testing with incomplete relevance judgments. In *Proceedings of the 16th ACM Conference on Information and Knowledge Management*, 2007.

[Carterette *et al.*, 2006] B. Carterette, J. Allan, and R. Sitaraman. Minimal test collections for retrieval evaluation. In *Proceedings of the 29th SIGIR Conference on Research and Development in Information Retrieval*, 2006.

[Chapelle and Chang, 2011] O. Chapelle and Y. Chang. Yahoo! Learning to rank challenge overview. *JMLR: Workshop and Conference Proceedings*, 14:1–24, 2011.

[Chapelle *et al.*, 2009] O. Chapelle, D. Metzler, Y. Zhang, and P. Grinspan. Expected reciprocal rank for graded relevance. In *Proceeding of the Conference on Information and Knowledge Management*, 2009.

[Herbrich *et al.*, 2000] R. Herbrich, T. Graepel, and K. Obermayer. Large margin rank boundaries for ordinal regression. *Advances in Large Margin Classifiers*, pages 115–132, 2000.

[Järvelin and Kekäläinen, 2002] K. Järvelin and J. Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM Transactions on Information Systems*, 20(4):422–446, 2002.

[Li *et al.*, 2007] P. Li, C. Burges, and Q. Wu. Learning to rank using classification and gradient boosting. In *Advances in Neural Information Processing Systems*, 2007.

[Long *et al.*, 2010] B. Long, O. Chapelle, Y. Zhang, Y. Chang, Z. Zheng, and B. Tseng. Active learning for ranking through expected loss optimization. In *Proceedings of the 33rd SIGIR Conference on Research and Development in Information Retrieval*, 2010.

[McCullagh, 1980] P. McCullagh. Regression models for ordinal data. *Journal of the Royal Statistical Society. Series B (Methodological)*, 42(2):109–142, 1980.

[Mohan *et al.*, 2011] A. Mohan, Z. Chen, and K. Weinberger. Web-search ranking with initialized gradient boosted regression trees. In *JMLR: Workshop and Conference Proceedings*, volume 14, pages 77–89, 2011.

[Radlinski and Joachims, 2007] F. Radlinski and T. Joachims. Active Exploration for Learning Rankings from Clickthrough Data. In *Proceedings of the 13th SIGKDD International Conference on Knowledge Discovery and Data Mining*, 2007.

[Sawade *et al.*, 2010] C. Sawade, S. Bickel, and T. Scheffer. Active risk estimation. In *Proceedings of the 27th International Conference on Machine Learning*, 2010.

[Sawade *et al.*, 2012a] C. Sawade, S. Bickel, T. von Oertzen, T. Scheffer, and N. Landwehr. Active evaluation of ranking functions based on graded relevance. In *Proceedings of the 23rd European Conference on Machine Learning*, 2012.

[Sawade *et al.*, 2012b] C. Sawade, N. Landwehr, and T. Scheffer. Active comparison of prediction models. In *Advances in Neural Information Processing Systems*, 2012.

[Zheng *et al.*, 2007] 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 *Advances in Neural Information Processing Systems*, 2007.