An Approach to Information Retrieval Based on Statistical Model Selection

Miles Efron*

August 15, 2008

Abstract

Building on previous work in the field of language modeling information retrieval (IR), this paper proposes a novel approach to document ranking based on statistical model selection. The proposed approach offers two main contributions. First, we posit the notion of a document's "null model," a language model that conditions our assessment of the document model's significance with respect to the query. Second, we introduce an information-theoretic model complexity penalty into document ranking. We rank documents on a penalized log-likelihood ratio comparing the probability that each document model generated the query versus the likelihood that a corresponding "null" model generated it. Each model is assessed by the Akaike information criterion (AIC), the expected Kullback-Leibler divergence between the observed model (null or non-null) and the underlying model that generated the data. We report experimental results where the model selection approach offers improvement over traditional LM retrieval.

Categories and Subject Descriptors: H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval–*retrieval models*; H.3.4 [**Information Storage and Retrieval**]: Systems and Software–*performance evaluation* (efficiency and effectiveness).

Keywords: Information retrieval models, Language modeling, Model selection, Akaike information criterion (AIC)

1 Introduction

In the context of applied statistics, the term *model selection* refers to the choice of a single model from a pool of candidate models. Most formal model selection techniques balance a measure of model goodness of fit

*Affiliation: School of Information, University of Texas. School of Information, 1 University Station, D7000, Austin, TX, 78712. email: miles@ischool.utexas.edu.

with a penalty for model complexity. During model selection we favor models that fit the data well. But to avoid overfitting, we employ Occam's razor and seek models that describe the data simply. Balancing these mandates–good fit and simplicity–has given rise to many methods of comparing models such as Mallows' C_p [Neter et al., 1996], the Bayesian information criterion [Schwarz, 1978] and the Akaike information criterion (AIC) [Burnham and Anderson, 2002].

In this paper we propose a novel method of ranking documents against a query during information retrieval (IR). The proposed approach, which we call model selection information retrieval (MSIR), ranks documents on a statistic closely related to AIC–the AIC difference between the document model and its corresponding null model.

Given a document d_i we derive a statistic corresponding to a test on the null hypothesis, H_0*: It is no more likely that the language model that generated document* d_i *generated the query than it is that a model generating terms at the rate we expect due to chance generated the query.* In this setup, each hypothesis, null and non-null is represented as a generative language model, and thus we rank documents on the evidence against the null using standard model selection criteria. Specifically, we rank documents on the difference in the Akaike information criterion (AIC) between the non-null and null models. AIC is the expected Kullback-Leibler divergence between a given model and the unknown model that generated the data. Thus ranking documents by AIC difference offers a theoretically sound method of conducting IR.

The approach we pursue shares the strengths of standard language modeling IR. But it also offers several benefits. The role of a null model is quite general. Using an explicit null model separates language model smoothing from inverse document frequency (IDF) term weighting. Because our approach decouples model estimation from IDF, the role of the null in the proposed framework also admits information such as relevance feedback or personalization data.

The second benefit of the proposed model selection IR theory lies in the application of AIC to document ranking. AIC balances the bias/variance tradeoff in model selection by considering two factors: model goodness of fit (i.e. the likelihood) and model complexity. In IR this translates to considering a language model's fit to the query while explicitly applying length normalization. An interesting result of this paper is that length normalization applies not only to documents, but to the number of query terms matched by each document. We argue that we can improve retrieval performance by mitigating the role of query-document term coordination. Instead of rewarding documents that match many query terms, we argue, we should reward documents that match the best query terms. Using AIC differences affords a natural means of operationalizing this intuition.

After Section 2's overview of our notation and experimental data sets, we turn to a review of language modeling IR. In section 4 we discuss a potentially harmful bias in the log-likelihood ratio that forms the

core of most probabilistic IR models. Our discussion here focuses on log-likelihood ratios in the context of language modeling. Section 5 outlines the model selection IR framework (MSIR). To test the merits and the motivation for MSIR, we report the results of several experiments in Section 6. Finally, Section 7 summarizes our findings and suggests avenues for future research in model selection-based IR.

2 Preliminaries

To clarify our exposition, Table 1 summarizes the notation used throughout our discussion. As we proceed, we will introduce a small amount of additional notation as needed to clarify our discussion.

Table 1: Notation used Throughout this Paper

$n(C)$	Grand total of word tokens in the collection (word count for C)
$n(w)$	Total number of times word w occurs in the collection
$n(D)$	Total number of documents in the collection
$n(d)$	Number of word tokens in document d (document length)
$v(d)$	Number of unique words in document d (i.e. the vocabulary)
$n(w;d)$	Number of times word w occurs in document d
$n(q)$	Number of word tokens in query q (query length)
$v(q)$	Number of unique words (i.e. the vocabulary) in query q
$m(q;d)$	Number of unique query words that are also in d (number of matches)

Throughout this paper will give examples and test hypotheses using data collected to support the National Institute for Standards and Technology's Text REtrieval Conference (TREC) [NIST,]. Each dataset consists of three parts: a corpus of documents, a collection of queries (called "topics" in TREC), and a list of all documents that are relevant to each query.

Table 2: Datasets Used for Experimentation

Corpus	# Docs	Text Type	TREC Topics	Name
LA	272,880	Newswire	351-400	LA.S (short); LA.L (long)
TREC-7	527,094	Mixed	351-400	TREC-7.S (short); TREC-7.L (long)
TREC-8	527,094	Mixed	401-450	TREC-8.S (short); TREC-8.L (long)
wt10g	1,692,096	Web data	401-450	wt10g.S (short); wt10g.L (long)

The LA dataset consists of Los Angeles Times articles from 1993-1994 and appears on TREC disk 5. TREC-7 and TREC-8 contain the data from TREC disks 4-5 minus the Congressional record; this is a mixture of news and governmental text. These corpora were used for the ad hoc retrieval tasks in the seventh and eighth TREC conferences. The wt10g dataset is a sample of the World Wide Web containing primarily HTML data.

As shown in the rightmost column of Table 2 we present results using two types of queries. Short queries use brief keyword statements of information need taken from the title field of each topic. Long queries utilize the title, description and narrative topic fields.

All experiments we report were undertaken using the Lemur IR toolkit [Project,]. During experimentation we did not apply any stoplists or stemming.

3 Standard Language Modeling IR

Methods based on statistical language modeling (LM) are among the most studied and the most successful modern information retrieval (IR) techniques [Ponte and Croft, 1998, Croft and (eds.), 2003, Zhai and Lafferty, 2004, Lafferty and Zhai, 2001, Berger and Lafferty, 1999]. The most common approach to language modeling information retrieval (LMIR) is the so-called unigram query generation model. Under this framework a probabilistic language model $\mathcal{D}_i$ is induced on each document in a corpus. Documents are then ranked in decreasing order of the likelihood that their corresponding models generated a user's query. The intuition is that a document whose model is likely to have generated the query has a pattern of word usage that is similar to the query's, and therefore it is likely that such a document is relevant to the query.

Usually $\mathcal{D}_i$ is a multinomial over the $v(d)$ unique words that occur in the document. Thus $\mathcal{D}_i$ is specified by a $v(d)$-vector of parameters $\boldsymbol{\theta}$ where θ_j is the probability that an observation from $\mathcal{D}_i$ generates word w_j.

Given a query and a corpus of documents, we rank the documents according to the likelihood that their corresponding models generated the query:

$$Pr(q|\mathcal{D}_i) = \prod_{w_j \in q} Pr(w_j|\mathcal{D}_i). \tag{1}$$

More typically, we rank by the log-likelihood

$$\log Pr(q|\mathcal{D}_i) = \sum_{w_j \in q} \log Pr(w_j|\mathcal{D}_i). \tag{2}$$

Here q is the set of words in the query.

In order to improve the ranking method of Eq. 2, the standard approach to language model-based IR relies on models that are smoothed in some fashion. Our discussion of model smoothing draws from the exposition offered in [Zhai and Lafferty, 2004].

Typically the smoothed model relies on two distributions. One distribution $Pr(w|\mathcal{D}_i^s)$ gives the probabilities for "seen" words–i.e. for query words that do appear in document d_i. The other distribution $Pr(w|\mathcal{D}_i^u)$

gives the probabilities for "unseen" words: words that appear in the query but are not in document d_i. This gives the log-probability

$$\log Pr(q|d_i) = \sum_{w_j \in q \cap d_i} \log Pr(w_j|\mathcal{D}_i^s) + \sum_{w_j \in q \cap \neg d_i} \log Pr(w_j|\mathcal{D}_i^u) \quad (3)$$

$$= \sum_{w_j \in q \cap d_i} \log \frac{Pr(w_j|\mathcal{D}_i^s)}{Pr(w_j|\mathcal{D}_i^u)} + \sum_{w_j \in q} \log Pr(w_j|\mathcal{D}_i^u). \quad (4)$$

Usually we take the probability of an unseen word to be proportional to the frequency of the word in the collection at large. Thus we have $Pr(w_j|\mathcal{D}_i^u) = \alpha Pr(w_j|C)$, where C indicates the "collection language model"–the probability of the word in the entire collection. The term α is a document-specific factor that ensures that $\boldsymbol{\theta}^u$ sums to one. This allows us to rewrite Eq. 4 as

$$RSV_{LM}(q; d_i) = \log Pr(q|d_i) = \sum_{w_j \in q \cap d} \log \frac{Pr(w_j|\mathcal{D}_j^s)}{\alpha Pr(w_j|C)} + v(d) \log \alpha \quad (5)$$

plus an addendum $\sum_{w_j \in q} \log Pr(w_j|C)$ that is constant for each query and thus does not affect ranking (we ignore it).

Equation 5 is the basic language modeling ranking function. We will refer to it often in the discussion below. We will refer to $\log Pr(q|d_i)$ as the retrieval status value (RSV) under the standard language modeling approach (LM).

3.1 Language Modeling Smoothing using Bayesian Bayesian Updating

The problem that remains is estimating $\mathcal{D}_i^s$, the smoothed language model for d_i. Though a variety of smoothing methods have been proposed, we focus on a single smoothing method in this paper–Bayesian updating. We limit discussion to a single smoothing method in the interest of brevity and clarity. We choose Bayesian updating because this method constitutes the state of the art in language modeling IR.

A multinomial language model has a Dirichlet distribution as its conjugate prior. Typically we choose the parameters of the Dirichlet to be proportional to the collection language model $Pr(w_j|C)$:

$$(\mu Pr(w_1|C),\ \mu Pr(w_2|C), \ldots, \mu Pr(w_{v(d)}|C)) \quad (6)$$

with μ being a hyperparameter whose magnitude governs the strength of the smoothing.

Having specified the prior distribution, the updated estimate for the probability of a matched (seen)

query word w_j in the model corresponding to document d_i is

$$\hat{\theta}(w_j|\mathcal{D}_i^s) = \frac{n(w_j;d_i) + \mu Pr(w_j|C)}{n(d) + \mu}. \tag{7}$$

In order to assure that all probabilities sum to 1, this leads to $\alpha = \frac{\mu}{n(d)+\mu}$.

Though other methods of smoothing exist, the results of this section give us all of the estimates we need in order to instantiate the query likelihood retrieval model shown in Eq. 5. Retrieval proceeds by selecting a value for the smoothing parameter μ. We then fit a smoothed language model to each document in the collection, ranking documents by their models' log-probability of having generated the query.

4 Bias in the Log-Likelihood Ratio

A log-likelihood ratio forms the basis of almost all probabilistic IR techniques [Roelleke and Wang, 2006]. The log-likelihood ratio between the document probability and the collection probability of a word is at the heart of the language modeling approach, as we see in Eq. 5. However, the log-likelihood ratio tends to exhibit a possibly problematic behavior in the context of IR. The simple log-likelihood ratio tends to be biased towards documents that match many query terms. Because $Pr(w_j|C)$ is almost always much smaller than the probability of w_j under the document model, the log-likelihood ratio tends to grow as more query terms are matched.

Rewarding documents that match query terms seems like a good idea. However, favoring high match counts is not necessarily to our advantage. Long queries, for instance, tend to have many words that are tangential to the query topic, and thus rewarding documents for matching many query words may inflate document scores based on weak evidence.

Figure 1 shows the distribution of matched query terms $m(q;d)$ in relevant and non-relevant documents in the LA dataset (long topics 351-400) and wt10g (long topics 401-450). The mean long query length for topics 351-400 is 57.5 words. The mean long query length for topics 401-450 is 51.66 words.

The figure suggests that on average relevant documents match slightly more terms than non-relevant documents do, but the difference between relevant and non-relevant documents is slim. The median number of matched terms for relevant LA documents is 15, with a median of 12 for non-relevant (standard deviation=7.046 for relevant and standard deviation=5.156 for non-relevant). The median number of matched terms for wt10g is 13 for both relevant and non-relevant documents (standard deviation=6.043 for relevant and standard deviation 6.081 for non-relevant). We found similar results for other data sets. It seems that we might like a small reward for the number of terms matched in a document, but that this reward should

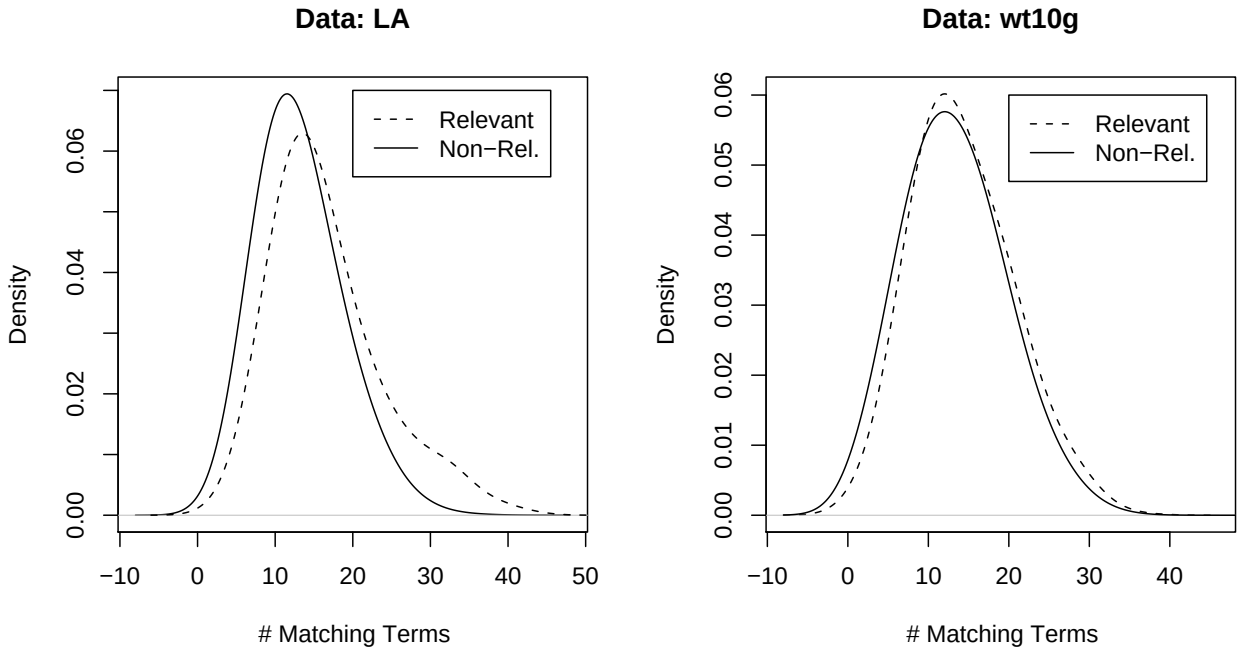

Figure 1: Distribution of # Matched Terms $m(q;d)$ for Two Data Sets

not be too great.

To understand how the log-likelihood's bias is related to language modeling IR let us re-write Eq. 5:

$$\log Pr(q|\mathcal{D}_i) = \sum_{w_j \in q \cap d_i} \log \frac{Pr(w_j|\mathcal{D}_i^s)}{\alpha Pr(w_j|C)} + v(q) \log \alpha \quad (8)$$

$$= \sum_{w_j \in q \cap d_i} \log \frac{Pr(w_j|\mathcal{D}_i^s)}{Pr(w_j|C)} + (v(q) - m(q;d)) \log \alpha \quad (9)$$

Eq. 9 shows that under the standard LM approach to retrieval, from the log-likelihood ratio we subtract a penalty for the number of query terms a document misses. Conversely, we reward documents proportionally to the number of query terms they match.

Given our earlier argument that the log-likelihood ratio can become inflated when a document matches many query terms, this reward seems risky; it will magnify what may already be a liability in the log-likelihood ratio. An interesting factor to consider in this light is the quantity α. It bears mentioning that in the case of Dirichlet smoothing where $\alpha = \frac{\mu}{\mu + n(d)}$, the penalty for missed terms will vary with document length.

The model selection approach to IR that we advance in the following discussion addresses the matter of bias from model complexity directly.

5 A Model Selection Approach to IR

We propose ranking documents on the difference in AIC between two language models. One model is the language model that generates query words at the rate responsible for creating the document. The other model generates query words at the rate corresponding to some null condition. In this paper we take the null condition to be the rate of query word generation that we expect due to chance. The difference in AIC between these two models amounts to a statistic on the hypothesis test:

> H_0: *It is no more likely that the document model generated the query than it is that the null model generated the query*
> H_1: *It is more likely that the document language model generated the query than it is that the null model generated the query.*

Instead of using the raw likelihoods to assess each model (null and non-null) we use the Akaike information criterion. AIC conditions model likelihood on a measure of model complexity. In the following discussion we argue that this conditioning mitigates the bias due to query-document coordination described above.

5.1 The Akaike Information Criterion (AIC) in Model Selection IR

To operationalize the comparison of each model's likelihood, we use the measure known as the Akaike information criterion (AIC) of each model [Akaike, 1973]. The AIC of a model $\mathcal{M}$ is

$$AIC(\mathcal{M}) = 2 \log \mathcal{L}(\mathcal{M}) - 2k \tag{10}$$

where $\mathcal{L}$ is the likelihood of the model given the data and k is the number of parameters in the model. Intuitively, AIC balances model goodness of fit with the principal of parsimony, avoiding models of unnecessarily high complexity.

AIC is one of many methods used to balance the bias/variance tradeoff in statistical model selection (cf. [Hastie et al., 2001]). In addition to AIC, measures such as Mallows' C_p [Neter et al., 1996], and the Bayesian information criterion [Schwarz, 1978] (to name only a few) condition a model's goodness of fit on its complexity. The intuition behind such measures is that we favor models that explain the data well. But, following Occam's razor, we prefer models that fit the data well without recourse to numerous parameters.

However, AIC is not simply a heuristic measure. In fact Akaike shows that -1 times AIC[1] is the expected Kullback-Leibler divergence between $\mathcal{M}$ and the unknown model that generated the data (the factor of 2 in AIC leads this to be true) [Burnham and Anderson, 2002]. Thus AIC offers a principled way to compare

[1] AIC is sometimes written differently so that 1 times AIC is the expected KL divergence.

the relative merits of two competing models. Given two models $\mathcal{M}_i$ and $\mathcal{M}_j$ we prefer the model with lower KL divergence (i.e. higher AIC) from the underlying model that generated the data.

With this result in mind, we define the scoring formula for the model selection IR approach:

$$RSV_{MSIR}(d_i, q) = AIC(\mathcal{D}_i) - AIC(\mathcal{N}_i). \tag{11}$$

Of course if our two models, null and non-null, have an equal number of parameters, then Eq. 11 is simply the log-likelihood ratio between them. However, in the following discussion we shall define our models in such a way that they are not of equal complexity.

5.2 Poisson Language Models

In this subsection we formalize our two models, null and non-null. Without altering the results of our previous discussion, it is convenient to define our models as independent Poisson language models, as opposed to the more typical multinomials. This is the case because Poisson language models relax the constraint that all parameters sum to 1. Mei, Fang, and Zhai offer a thorough discussion of Poisson language models for IR in [Mei et al., 2007]. The following discussion draws on their exposition.

Whereas the multinomial is specified by a parameter vector $\boldsymbol{\theta}$, giving the probabilities of each word in the vocabulary V, the independent Poisson model is specified by a vector $\boldsymbol{\Lambda}$, whose elements are the rates at which each word is generated per unit of text.

Let d_i be a piece of text (a document or a query). If t is a unit of time taken to write d_i, we assume that the count of a word w_j in d_i follows a Poisson distribution with mean $\lambda_j t$, where λ_j gives the expected number of w_j seen per unit of time. Thus the probability of w_j follows

$$Pr(c(w_j, d_i) = k|\lambda_j) = \frac{\exp[-\lambda_j t](\lambda_j t)^k}{k!}. \tag{12}$$

Given a vocabulary of V words, we have V corresponding independent Poisson processes that comprise $\boldsymbol{\Lambda} = \{\lambda_1, \ldots, \lambda_V\}$. The parameter vector $\boldsymbol{\Lambda}$ and the Poisson density constitute a Poisson language model.

With this setup, the maximum likelihood estimator for λ_j in document d_i is simply $\frac{n(w_j;d_i)}{n(d_i)}$, precisely the estimate for the corresponding multinomial.

Standard smoothing methods also translate easily into the Poisson framework. Jelinek-Mercer smoothing operates in the same fashion for Poissons that it does for multinomials. Bayesian updating for Poissons uses the gamma distribution instead of the Dirichlet. However, it has been shown that the log-probabilities derived for a hyperparater μ are the same under a Dirichlet-smoothed multinomial and a Gamma-smoothed

Poisson. Using Bayesian updating we have the rate of a seen word under a Poisson language model for document d_i:

$$\hat{\lambda}_{w,i} = \frac{n(w; d_i) + \mu Pr(w_j|C)}{n(d_i) + \mu} \tag{13}$$

which is the same as the probability for a seen word w_j under the multinomial.

So long as we do not deviate from these estimation methods, by using Poisson language models our results will be the same as we expect from multinomial models. The only difference germane to our discussion is that a Poisson language model's parameters need not sum to 1.

5.3 Null and Non-Null Models

In this section we define the models that comprise the null and non-null conditions for the model comparison operation described earlier. Let $m(q; d_i)$ be the number of query words matched by document d_i. Both models, null and non-null, are Poisson language models with $m(q; d_i)$ dimensions, defined over each matching query word. This is in contrast to the standard LMIR approach where the document language model is specified over the $v(d_i)$ documents in d_i.

Before beginning our definitions a high-level comment will make things clearer. The goal in the following discussion is to induce for each document d_i a *pair* of language models. One language model is very similar to the models induced during standard LMIR. The other model, the null model, is of the same dimension as the document model. But the null model generates text at a rate corresponding to a null condition. For simplicity we take the null condition as equal to the rate of generation in the corpus at large.

If we assume that the null model generates matched query terms at the rate we would expect due to chance, the entire collection is an obvious source of data for the null model. Thus we estimate the null rate parameter for word w_j with $\lambda_{j0} = \frac{n(w;C)}{n(C)}$. The log-likelihood that the null model corresponding to document d_i generated the query is thus

$$\log Pr(q|\mathcal{N}_i) = \sum_{w_j \in q \cap d_i} \log Pr(w_j|\mathcal{N}_i) \tag{14}$$

$$= \sum_{w_j \in q \cap d_i} \log Pois(n(w_j; q)|\lambda_{j,0} n(q)) \tag{15}$$

The non-null model is defined as the language model corresponding to the document under consideration. The MLE for word w_j's rate in document d_i's model is, of course the number of times w_j occurs in d_i divided by the length of d_i. However, in practice we will smooth the document model using Bayesian updating with Gamma priors.

Eq. 13 gives our estimated rate parameter for word w_j in document d_i. This leads to the log-probability

of the query q under the document model:

$$\log Pr(q|\mathcal{D}_i) = \sum_{w_j \in q \cap d_i} \log Pr(w_j|\mathcal{D}_i) \quad (16)$$

$$= \sum_{w_j \in q \cap d_i} \log Pois(n(w_j;q)|\hat{\lambda}_{j,i} n(q)) \quad (17)$$

Because our models, null and non-null, are defined over *seen* query terms, the log-probability of the query given $\mathcal{D}_i$ (our smoothed model) is simply a sum over the log-likelihoods, as opposed to the quantity shown in Eq. 5 which allocates likelihood over unseen words as well.

5.4 Interpreting the Models

The model $\mathcal{D}_i$ is the language model that generated the expression of query terms in document d_i. Throughout our experimental evaluation we assume that $\mathcal{D}_i$ is a smoothed estimate. However, under the MSIR framework we need make no assumption that the reference model used for smoothing is the null model. Thus MSIR decouples smoothing in service to document model estimation and the influence of inverse document frequency (IDF).

The query likelihood on $\mathcal{D}_i$ quantifies the similarity between the query and the document. However, it is not the raw magnitude of $\mathcal{L}(\mathcal{D}_i)$ that interests us. Rather it is the magnitude of the likelihood with respect to our null condition. In this paper the null condition corresponds to the hypothesis that $\mathcal{D}_i$ generates query terms that the rate we expect due to chance. Thus we take the rate parameters for the null model $\mathcal{N}_i$ as equal to the rates in the collection language model C.

Equating our null condition with the likelihood of the collection language model invites a comparison between MSIR and the divergence from randomness model proposed in [Amati and Rijsbergen, 2002] and [Amati, 2006]. In both MSIR and divergence from randomness, a document's RSV depends on the observed difference between the observed likelihood and the likelihood expected due to chance. However, two points distinguish MSIR from the divergence from randomness model. First, MSIR depends on query generation likelihoods, while divergence from randomness is concerned with document generation probabilities. Secondly, choosing the background rate as our null is only one method of proceeding under MSIR; a different null would mitigate the similarity between MSIR and divergence from randomness.

The log-likelihood ratio $\log \frac{Pr(q|\mathcal{D}_i)}{Pr(q|\mathcal{N}_i)} = \log Pr(q|\mathcal{D}_i) - \log Pr(q|\mathcal{N}_i)$ can be viewed as a test statistic on the null hypothesis that $\mathcal{D}_i$ and $\mathcal{N}_i$ have equal likelihood with respect to the query. However our test statistic has a problem if we wish to compare its magnitude on a document d_i to the statistic derived for another document d_j. Since $m(q;d_i)$ may not equal $m(q;d_j)$ the likelihoods are not directly comparable. From

a model selection standpoint, $\mathcal{D}_i$ may be of different complexity than $\mathcal{D}_j$, making a comparison of their likelihoods ill posed. Model selection theory counsels us to "normalize" likelihood on model complexity before comparing goodness of fit. This is the role that AIC's penalty for model degrees of freedom plays in our discussion.

5.5 Model Degrees of Freedom

Since the null model and the non-null model are both defined over $m(q; d_i)$ terms, it is intuitive that they should have the same number of parameters, and thus the difference in their AIC's would reduce to the simple log-likelihood between them. However, in this section we define each model's degrees of freedom in a way that takes document length and query length into account.

With respect to the non-null model $\mathcal{D}_i$, the obvious estimate for the number of free parameters is the dimension of the model, $m(q; d_i)$. However, we assume that $m(q; d_i)$ is a statistic that corresponds to an unknown parameter M. A document similar to d_i might show a different observed number of matched query terms. Thus we take the number of parameters in $\mathcal{D}_i$ to be the expected value of M, that is, the expected number of matched query terms given q and d_i.

To find this expectation we begin by considering the random variable X, the number of *missed* query terms in d_i. The expected value of X given q and d_i is

$$E(X) = \sum_{w_j \in q} (1 - \frac{n(w_j)}{n(C)})^{n(d_i)}. \tag{18}$$

Thus $E(M)$, the expected number of matching terms, is $v(q) - E(X)$. We take the degrees of freedom of the non-null model to be $E(M)$, which is a function of the query and the document length.

We take the degrees of freedom in the null model to be $v(q)$. Our motivation in this choice is twofold. First, we assume that the null models for all documents will have very low likelihood and will thus be comparable. Therefore defining the null degrees of freedom as a query-specific constant makes sense. Additionally, our number of non-null degrees of freedom (Eq. 18) is computed over all $v(q)$ query terms, inviting a $v(q)$-dimensional null.

These definitions lead to our final definition of the model selection RSV

$$RSV_{MSIR}(q; d_i) = \log \mathcal{L}(\mathcal{D}_i) - E(M) - \log \mathcal{L}(\mathcal{N}_i) - v(q) \tag{19}$$

which is the difference in AIC between the null and non-null language models.

5.6 Relationship between LM and MS Retrieval Status Values

To understand the close relationship between the standard LM approach and the MSIR technique developed here we may rewrite Eq. 19

$$RSV_{MSIR}(q; d_i) = \sum_{w_j \in q \cap d_i} \log \frac{Pr(w_j|\mathcal{D}_j)}{Pr(w_j|\mathcal{N}_i)} + v(q) - E(M). \tag{20}$$

In this paper we have defined the parameters of the null model in terms of the collection language model C. Thus, if we compare this quantity with Eq. 5 we see that both the standard LM approach and MSIR yield a retrieval status value that can be written

$$RSV_P(q; d_i) = \sum_{w_j \in q \cap d_i} \log \frac{Pr(w_j|\mathcal{D}_j)}{Pr(w_j|C_i)} + P_{q,i} \tag{21}$$

where $P_{q,i}$ is a penalty for document d_i on query q. The difference between the two methods, then, lies in the penalty that is applied to the log-likelihood ratio. If we omit consideration of smoothing-specific penalization, the standard LM approach penalizes documents for each query term d_i fails to match. However, smoothing methods such as Bayesian updating will bring document length into play. The penalty in MSIR depends on $E(M)$, which depends on document length and query length but not directly on $m(q; d_i)$ (Eq. 18).

The similarity we have just derived stems from our tight coupling of the null condition and the collection language model. This corresponds to a null hypothesis that the non-null model is no more likely to have generated the query than a model that generates terms at the rate we expect due to chance. Defining the null in terms of the collection probabilities makes sense and will make the following evaluation easy to interpret. However, it is important to note that the null model need not be derived from the collection language model. As an example of an alternative, we might consider a case where we have relevance feedback information. Here we might consider the null as a mixture between the collection language model and a language model that emits query words at a rate inversely proportional to the user's relevance model.

6 Experimental Results

We undertook a series of experiments to test the proposed model selection IR framework. During experimentation our goals were twofold. We aimed to test whether MSIR offers improvement over standard LM-based IR. We reason that the proposed framework is itself interesting. However, we also hypothesized that using AIC differences to rank documents would improve retrieval over standard LMIR due to an improved penalty for model complexity. Besides a standard evaluation of IR effectiveness, then, we addressed the question:

does model complexity detract from the effectiveness of standard LM-based retrieval?

In the experiments that follow we test the model selection retrieval approach described above, using Eq. 19 as our ranking function. As a baseline of comparison we test MSIR against standard language modeling IR, using Eq. 5 as described in [Zhai and Lafferty, 2004]. We abbreviate this baseline to LM.

To simplify our analysis we have tested each model using only a single form of smoothing: Bayesian updating. Thus standard LMIR uses Dirichlet smoothing and MSIR is tested using Gamma smoothing. We have tested each method using a variety of smoothing parameters as discussed below.

6.1 Retrieval Effectiveness Using Model Selection IR

One of our main arguments is that using AIC differences between document models and null models offers an improvement over standard language modeling IR. To test this hypothesis we performed *ad hoc* retrieval experiments using the data and queries shown in Table 2.

Summary statistics from these experiments appear in Tables 3-6. These tables report results obtained from runs using LM and MSIR. Each row of the tables corresponds to a dataset/query set pairing (i.e. a retrieval run). Columns correspond to observed retrieval effectiveness measures. We report four measures, mean average precision (MAP) taken over 11 recall levels, precision at 5 documents retrieved (P5), precision at 10 documents retrieved (P10) and R-precision [Aslam and Yilmaz, 2005, Buckley and Voorhees, 2000]. R-precision is precision observed at r documents retrieved, where r is the number of documents relevant to a query. Overviews of precision and recall-based evaluation metrics are available in [Manning et al., 2008, Losee, 2000, Cleverdon and Mills, 1963].

To give a sense of the role that smoothing plays in our experiments we report results at three levels of smoothing: low ($\mu = 100$), medium ($\mu = 2000$), and high ($\mu = 10000$).

Italicized entries indicate that MSIR performed worse than LM. Runs in which MSIR performed better than LM at a statistically significant level are marked with a $+$. Runs marked with a $-$ indicate that MSIR was significantly worse than LM. Statistical significance is measured here using the Wilcoxon rank sum test (paired and one-directional) with a p-value less than 0.05.

Tables 3-6 report results from a total of 96 experimental conditions (8 corpus/query pairs, three levels of smoothing, four effectiveness measures). In 28 of these instances, MSIR performed significantly better than the standard language modeling approach. MSIR was significantly worse on 3 conditions.

Of particular interest is the performance observed at low levels of smoothing. When μ was low, MSIR performed decisively better than standard LMIR. This makes sense because the penalty applied to the log-likelihood ratio under MSIR does not depend on the smoothing-specific α. The relatively poor performance

Table 3: Retrieval Effectiveness (MAP) of Standard Language Modeling (LM) and Model Selection IR (MSIR)

	Low Smoothing		Medium Smoothing		High Smoothing	
	LM	MS	LM	MS	LM	MS
LA.S	0.1669	0.1774+	0.2244	*0.2135*	0.2141	*0.2139*
LA.L	0.1346	0.2164+	0.2401	0.2417	0.2378	*0.2342*
TREC-7.S	0.1657	0.1710	0.1891	*0.1877*	0.1860	0.1866
TREC-7.L	0.1362	0.1871+	0.2087	0.2089	0.2099	0.2109
TREC-8.S	0.2010	0.2040	0.2106	*0.2100*	0.1950	*0.1942*
TREC-8.L	0.1751	0.2094+	0.2274	0.2315+	0.2133	0.2272+
wt10g.S	0.1229	0.1229	0.1686	0.1693	0.1655	0.1678
wt10g.L	0.0836	0.1099+	0.1638	0.1642	0.1630	*0.1625*

Table 4: Retrieval Effectiveness (Precision at 10) of Standard Language Modeling (LM) and Model Selection IR (MSIR)

	Low Smoothing		Medium Smoothing		High Smoothing	
	LM	MS	LM	MS	LM	MS
LA.S	0.2360	0.2400	0.2960	0.2960	0.2760	0.2780
LA.L	0.2300	0.3000+	0.3420	*0.3300*	0.3300	*0.3200*
TREC-7.S	0.3740	0.3780	0.4300	0.4320	0.4080	0.4100
TREC-7.L	0.3700	0.4200+	0.4740	*0.4640*	0.4460	0.4480
TREC-8.S	0.4340	0.4440+	0.4260	*0.4240*	0.3760	0.3780
TREC-8.L	0.3580	0.4460+	0.4280	0.4380	0.3860	0.4020+
wt10g.S	0.2104	*0.2062*	0.2354	0.2417	0.2354	*0.2312*
wt10g.L	0.1580	0.2060	0.3020	*0.2820-*	0.2660	*0.2620*

Table 5: Retrieval Effectiveness (Precision at 5) of Standard Language Modeling (LM) and Model Selection IR (MSIR)

	Low Smoothing		Medium Smoothing		High Smoothing	
	LM	MS	LM	MS	LM	MS
LA.S	0.2800	*0.2720*	0.3440	0.3600+	0.3480	0.3600
LA.L	0.2840	0.3480+	0.3880	0.4000	0.3960	*0.3680-*
TREC-7.S	0.4200	*0.4080-*	0.4720	0.4720	0.4400	*0.4360*
TREC-7.L	0.3920	0.4600+	0.5040	0.5080	0.5200	0.5200
TREC-8.S	0.4720	0.4880+	0.4840	*0.4760*	0.4440	*0.4320*
TREC-8.L	0.4160	0.4760+	0.4640	0.4680	0.4280	0.4720+
wt10g.S	0.2500	*0.2458*	0.2792	*0.2917*	0.2875	0.2875
wt10g.L	0.1640	0.2560+	0.3240	*0.3160*	0.3320	*0.3200*

Table 6: Retrieval Effectiveness (R-precision) of Standard Language Modeling (LM) and Model Selection IR (MSIR)

	Low Smoothing		Medium Smoothing		High Smoothing	
	LM	MS	LM	MS	LM	MS
LA.S	0.1792	0.2013+	0.2495	*0.2309*	0.2224	*0.2150*
LA.L	0.1559	0.2381+	0.2591	*0.2570*	0.2624	*0.2479*
TREC-7.S	0.2148	0.2238+	0.2465	*0.2437*	0.2380	0.2387
TREC-7.L	0.1837	0.2387+	0.2571	0.2576	0.2580	0.2590
TREC-8.S	0.2502	*0.2494*	0.2524	*0.2495*	0.2304	0.2347
TREC-8.L	0.2240	0.2542+	0.2744	0.2785+	0.2603	0.2816+
wt10g.S	0.1511	*0.1486*	0.2109	*0.2039-*	0.2006	0.2011
wt10g.L	0.1074	0.1239+	0.1899	*0.1847*	0.1871	0.1978+

of LM under low smoothing suggests that when μ is small Bayesian updating applies an improper penalty to the log-likelihood ratio.

In the case of low smoothing, the LM penalty will be heavily driven by document length. This raises the more general question of how a document length-based penalty fits into the scheme of retrieval performance. To see if a length-based penalty would improve the performance of LM (thus mitigating the gains seen using MSIR) we tested retrieval using length-based priors as described by Losada and Azzopardi [Losada and Azzopardi, 2008]. Most language modeling IR assumes uniform document priors. However, Losada and Azzopardi suggest using a non-uniform prior, where a document's prior probability is given by

$$Pr(d_i) = \frac{n(d_i)}{n(C)}.$$

This leads to a factor of $\log Pr(d_i)$ being added to $\log Pr(q|d_i)$ during document ranking. Table 7 shows retrieval effectiveness (as measured by MAP) using standard language modeling with non-uniform document priors (NUDP) and model selection retrieval (MSIR). For the experiments summarized in Table 7 we smoothed models using Bayesian updating with $\mu = 2000$.

Table 7: Retrieval Effectiveness (MAP) of Non-Uniform Document Priors (NUDP) and Model Selection IR (MSIR)

	MAP		P10	
	NUDP	MSIR	NUDP	MSIR
LA.S	0.2027	0.2135+	0.2700	0.2960+
LA.L	0.2422	*0.2417*	0.3520	*0.3300*-
TREC-7.S	0.1703	0.1877+	0.3900	0.4320+
TREC-7.L	0.2086	0.2089	0.4760	*0.4640*
TREC-8.S	0.1857	0.2100	0.3840	0.4240+
TREC-8.L	0.2222	0.2315	0.4260	0.4380
wt10g.S	0.1329	0.1693+	0.2208	0.2417
wt10g.L	0.1634	0.1642	0.3000	*0.2820*

By using AIC differences to condition its model likelihoods, MSIR induces a penalty that depends directly on document length (Eq. 18). Interestingly, MSIR penalizes long documents, while the method pursued in [Losada and Azzopardi, 2008] rewards longer documents. Table 7 shows an advantage to MSIR; we see that MSIR enjoys stronger performance relative to the (non-uniform) LM model than it saw versus the LM model with uniform priors (Table 3).

6.2 Effect of Model Smoothing

Tables 3-6 suggest that a key difference between standard language modeling IR and MSIR lies in each approach's sensitivity to model smoothing. Figures 2 and 3 give more detail on the relationship between smoothing and performance under each approach.

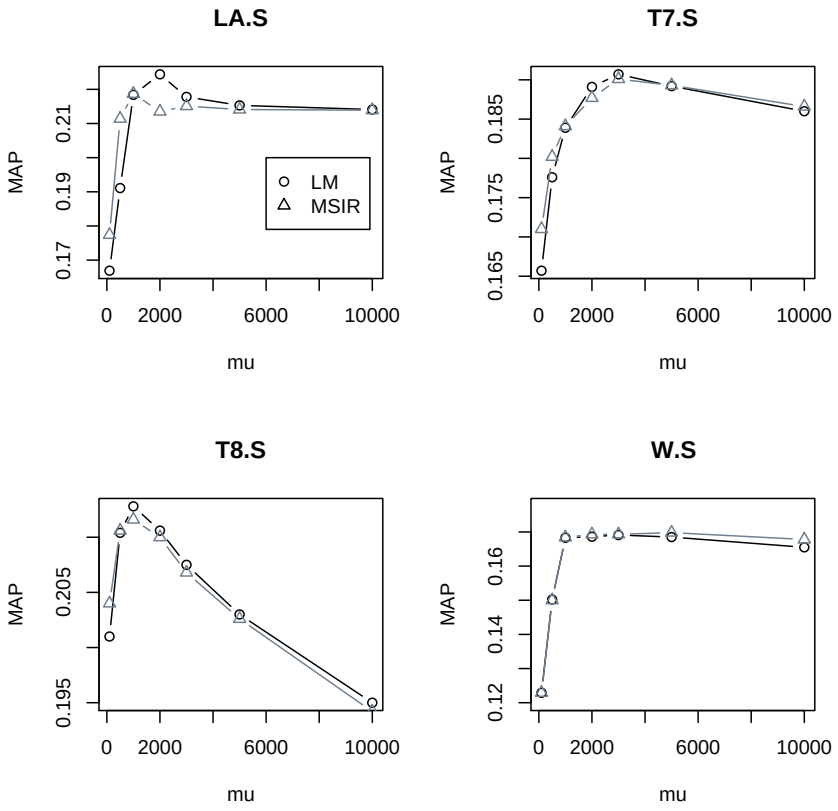

Figure 2: Effect of Smoothing Parameter on MAP Retrieval Performance (Short Queries)

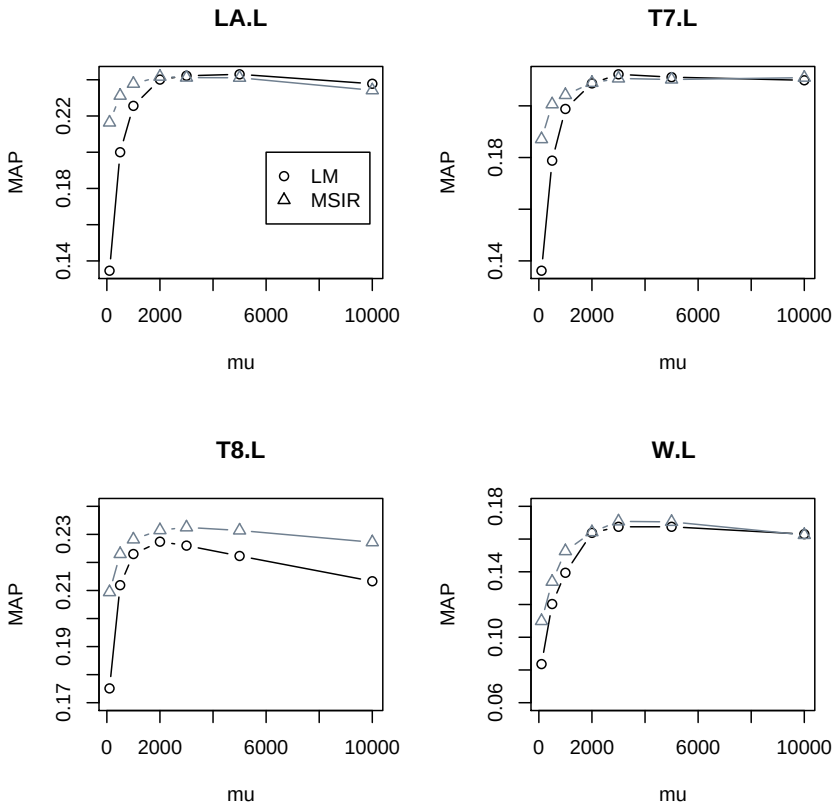

Figure 3: Effect of Smoothing Parameter on MAP Retrieval Performance (Long Queries)

The x-axis of Figure 2 shows mean average precision for short queries. The y-axis is μ, the Bayesian updating smoothing parameter. In this case it appears that MSIR and LM retrieval operate at similar levels of effectiveness as μ changes. However, in Figure 3 we see that MSIR is notably stronger than LM at low levels of smoothing. These results suggest that MSIR offers a level of safety with respect to smoothing; MSIR is rarely worse than LM and is often significantly better. In contexts where we lack relevance judgments to tune μ this safety would be very appealing.

6.3 Relationship Between LM and MSIR Retrieval Status Values

The results we have reported show very similar retrieval effectiveness for LM and MSIR in many experimental conditions. The similarity we see at middling and high levels of smoothing is somewhat surprising given the different penalties applied under each technique. Nonetheless, the similarities in retrieval effectiveness beg the question, is MSIR yielding a substantively different RSV than standard LM? Of course the results shown in Tables 3-6 demonstrate statistically significant differences between the methods. However, we can glean more detail on the relationship between the models' RSVs by looking at Figures 4-6. In these figures we plot the RSV obtained by model selection IR against the RSV from standard LM on nine queries (long topics 451-459 run against the wt10g data). These topics and data were selected because they are representative of the patterns we found across a wide spectrum of query/data plots.

Figure 4 shows the RSVs with a middling level of smoothing ($\mu = 2000$). We can see that some queries (e.g. 451 and 455) give RSVs that are quite similar under both IR models. Other topics (e.g. 453 and 456) exhibit different RSV distributions under each model.

Figures 5 and 6 plot RSVs obtained from runs using low and high smoothing, respectively. It is interesting to note that under both low and high smoothing conditions, the relationship between MSIR and LM's scores is much looser than it is under moderate smoothing. While we expected to see such a difference under light smoothing, we were surprised to find a similar difference using strong smoothing. We plan to pursue this issue in future work, along with an analysis of the performance obtained for topics with high correlation between MSIR and LM documents scores, versus topics with low correlation.

Returning to the question of how strongly related the scores given by MSIR and LM are, we offer a point of comparison. The correlation between the RSV obtained by MSIR and LM on the wt10g data with long queries was 0.796. The correlation between LM and Okapi [Robertson and Walker, 1994, Robertson et al., 1995] (with default weights from the Lemur toolkit) on the same data and queries was 0.761. This suggests that MSIR's retrieval status value is not much more correlated with LM than the RSV from a qualitatively different IR model.

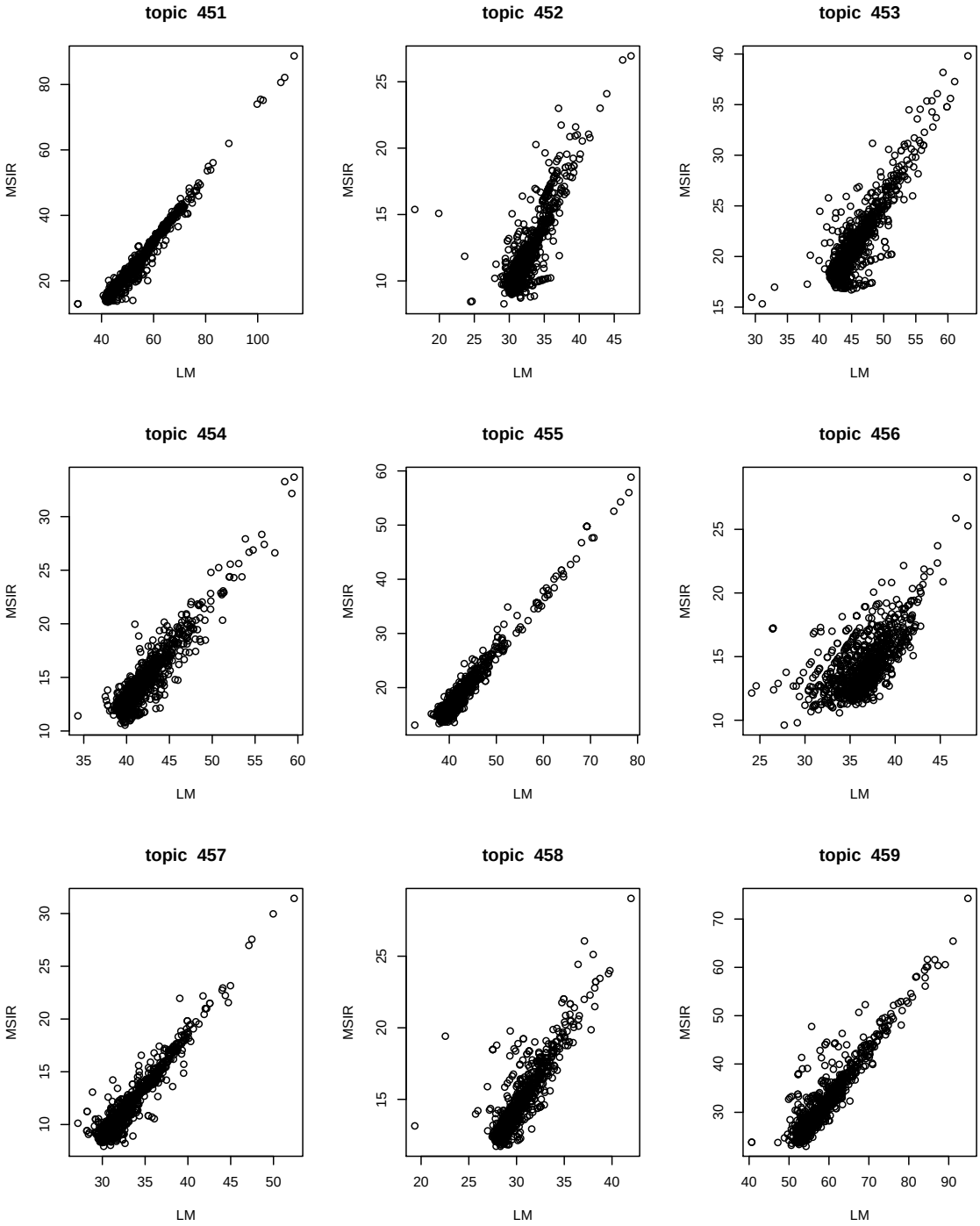

Figure 4: Relationship Between Language Modeling and Model Selection RSVs (W.L data, $\mu = 2k$)

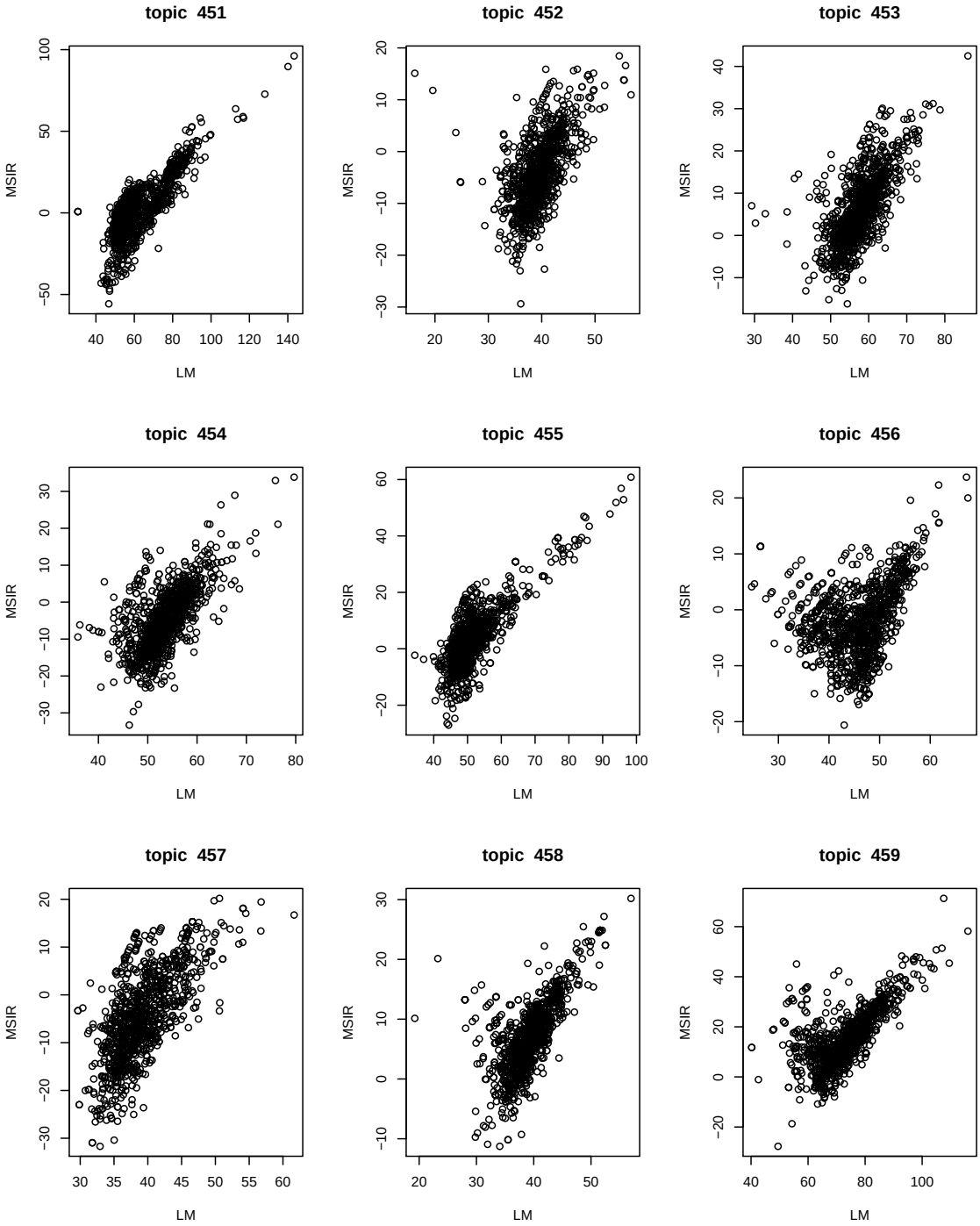

Figure 5: Relationship Between Language Modeling and Model Selection RSVs (W.L data, $\mu = 100$)

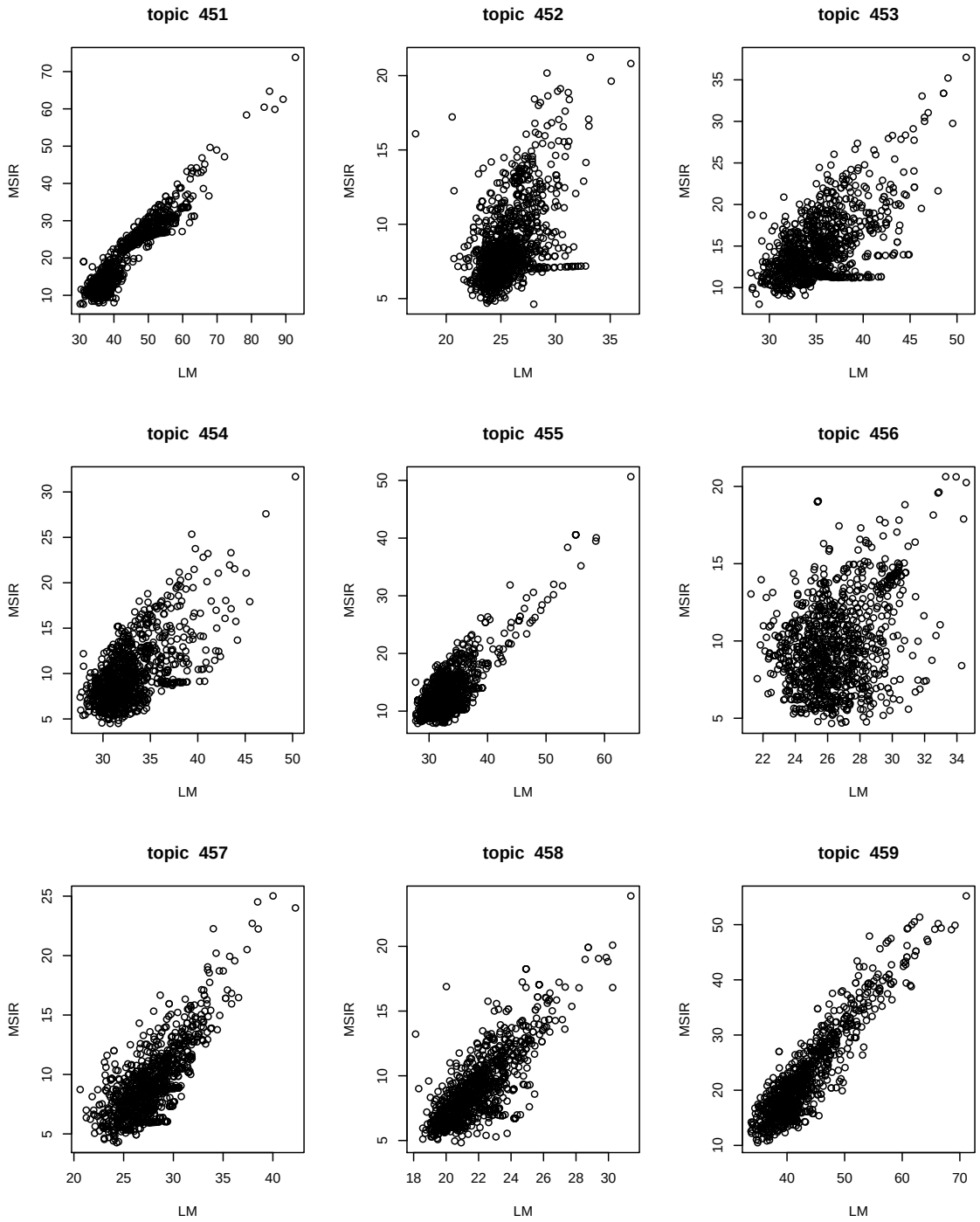

Figure 6: Relationship Between Language Modeling and Model Selection RSVs (W.L data, $\mu = 10k$)

6.4 RSV Correlation with Query-Document Coordination

A principle argument we have made is that language models should be penalized for the number of query terms matched by a document. We make this argument in efforts to find an RSV that is uncorrelated (or weakly correlated) with $m(q; d_i)$. This translates to conditioning model likelihoods on the complexity they incur. We have argued that using AIC differences to rank documents will reduce the bias in the log-likelihood ratio towards documents with high query term coordination.

Table 8: Correlations Between RSV's and Number of Terms Matched

	Low Smoothing		Medium Smoothing		High Smoothing	
	MS	LM	MS	LM	MS	LM
LA.L	0.184	-0.376	0.307	0.364	-0.014	0.496
TREC-7.L	0.059	-0.206	0.228	0.311	-0.027	0.398
TREC-8.L	-0.044	-0.205	0.126	0.269	-0.137	0.361
wt10g.L	*-0.216**	-0.063	-0.0512	0.270	-0.253	0.334

Table 8 helps us understand the relationship between the number of terms a document matches and the document's retrieval status value. The table shows observed correlations between the number of matched terms and the RSV obtained under the standard LM approach and using our model selection approach. The correlations were obtained by iterating over each of the 50 queries applied to each of our data sets. During each iteration we calculated the coefficient of correlation between each document's RSV using each model and the number of query words the document matched. The cell values contain the mean correlation, averaged over 50 queries. The table shows results for three levels of smoothing.

In all cases but two (italicized), MSIR yields a statistic that is less correlated with term coordination than LM. In all cases except the asterisked one, MSIR was significantly closer to zero than LM (using a one-tailed t-test with $p = 0.01$ as the cutoff for significance). In many cases we can see that the correlation between MSIR's retrieval status value and document-query coordination is very close to zero.

The correlation between RSV and coordination under MS also appears less related to smoothing than the standard language modeling's RSV. When $\mu = 100$ we can see that LM yields strongly negative RSV-coordination correlations (with the exception of the wt10g data). This is understandable since the document penalty under LM with Dirichlet priors is length-dependent. With a low μ, a Dirichlet-smoothed LM approach would favor long documents. However, documents are also penalized for low coordination. On average, long documents have a better chance of scoring a high coordination. Thus in the case of low Dirichlet smoothing the document length penalty and the coordination reward work at cross purposes. This result suggests why the MS approach outperforms standard LM retrieval for low values of μ.

7 Summary and Conclusion

This paper has proposed a novel approach to document ranking based on statistical model selection. For each document d_i in our collection we define a language model and a null model over the query words that appear in d_i. We then rank documents by the difference in AIC between the non-null and null models with respect to the query. This amounts to ranking documents on a statistic that gauges the evidence against the null hypothesis H_0 : *it is no more likely that the model that generated* d_i *also generated the query than it is that null model generated the query.* In this paper we take the null model to be the collection model, though this is not necessary, and exploring alternative nulls will occupy future research.

AIC is the expected Kullback-Leibler divergence between a model $\mathcal{M}$ and the unknown model that generated the data. Thus ranking documents by the difference in AIC between the non-null model $\mathcal{D}_i$ and the null model $\mathcal{N}_i$ provides a novel, principled extension of the language modeling approach to information retrieval. The experimental results reported in this paper suggest that model selection IR offers significant benefit to retrieval across a variety of experimental conditions.

Our contributions in this paper include

- A theoretical extension to language modeling IR based on statistical model selection
- An argument that RSV's should be uncorrelated or weakly correlated with query-document term coordination $m(q; d_i)$
- An experimental evaluation of the proposed model selection IR technique.

Our experiments suggest that MSIR is a promising method of ranking documents. We found that MSIR rarely performed significantly worse than LM. Additionally, MSIR frequently performed at a level significantly better than LM, especially when a small amount of smoothing was applied to language models.

In subsequent work, we plan to follow several avenues of research related to MSIR. For instance, in this paper we have equated the null model with the collection language model. However, we argue that the idea of a null model is quite flexible. In future work we will test methods for applying different sources of evidence and different semantics to the null model. We hypothesize that such an approach will allow MSIR to admit relevance feedback and personalization information retrieval. Additionally, in the interests of brevity we have limited our analysis here to language models smoothed with Bayesian updating. While Bayesian updating constitutes the state of the art in language modeling IR, it will be of interest to pursue the relationship between MSIR and other smoothing methods, such as Jellinek-Mercer and two-stage smoothing [Zhai and Lafferty, 2002, Zhai and Lafferty, 2006]. Finally, we plan to pursue methods of improving the model selection retrieval approach described here. While Section 5 gives a strong motivation for MSIR, we

suspect that pursuing alternative interpretations of model degrees of freedom and language model definitions will reward future research.

References

[Akaike, 1973] Akaike, H. (1973). Information theory as an extension of the maximum likelihood principle. In Petrov, B. N. and Csaki, F., editors, *Second International Symposium on Information Theory*, pages 267–281. Budapest.

[Amati, 2006] Amati, G. (2006). Frequentist and bayesian approach to information retrieval. In *Proceedings of the European Conference on Information Retrieval (ECIR 2006)*, pages 13–24, Berlin. Springer.

[Amati and Rijsbergen, 2002] Amati, G. and Rijsbergen, C. J. V. (2002). Probabilistic models of information retrieval based on measuring the divergence from randomness. *ACM Transactions on Information Systems (TOIS)*, 20(4):357–389.

[Aslam and Yilmaz, 2005] Aslam, J. A. and Yilmaz, E. (2005). A geometric interpretation and analysis of R-precision. In *CIKM '05: Proceedings of the 14th ACM international conference on Information and knowledge management*, pages 664–671, New York, NY, USA. ACM.

[Berger and Lafferty, 1999] Berger, A. and Lafferty, J. (1999). Information retrieval as statistical translation. In *SIGIR '99: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 222–229, New York, NY, USA. ACM.

[Buckley and Voorhees, 2000] Buckley, C. and Voorhees, E. M. (2000). Evaluating evaluation measure stability. In *SIGIR '00: Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 33–40, New York, NY, USA. ACM.

[Burnham and Anderson, 2002] Burnham, K. P. and Anderson, D. R. (2002). *Model Selection and Multimodel Inference*. Springer, New York, 2nd edition.

[Cleverdon and Mills, 1963] Cleverdon, C. W. and Mills, J. (1963). The testing of index language devices. *ASLIB Proceedings*, 15(4):106–130.

[Croft and (eds.), 2003] Croft, W. B. and (eds.), J. L. (2003). *Language Modeling for Information Retrieval*. Kluwer, Dordrecht.

[Hastie et al., 2001] Hastie, T., Tibshirani, R., and Friedman, J. H. (2001). *The Elements of Statistical Learning*. Springer.

[Lafferty and Zhai, 2001] Lafferty, J. and Zhai, C. (2001). Document language models, query models, and risk minimization for information retrieval. In *SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 111–119, New York, NY, USA. ACM.

[Losada and Azzopardi, 2008] Losada, D. E. and Azzopardi, L. (2008). An analysis on document length retrieval trends in language modeling smoothing. *Information Retrieval*, 11:109–138.

[Losee, 2000] Losee, R. M. (2000). When information retrieval measures agree about the relative quality of document rankings. *J. Am. Soc. Inf. Sci.*, 51(9):834–840.

[Manning et al., 2008] Manning, C. D., Raghavan, P., and Scheutze, H. (2008). *Introduction to Information Retrieval*. Cambridge University Press, Cambridge.

[Mei et al., 2007] Mei, Q., Fang, H., and Zhai, C. (2007). A study of poisson query generation model for information retrieval. In *SIGIR '07: Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 319–326, New York, NY, USA. ACM.

[Neter et al., 1996] Neter, J., Kutner, M. H., Nachtsheim, C. J., and Wasserman, W. (1996). *Applied Linear Statistical Models*. Irwin, Chicago, 4th edition.

[NIST,] NIST. The Text REtrieval Conference (TREC), http://trec.nist.gov.

[Ponte and Croft, 1998] Ponte, J. M. and Croft, W. B. (1998). A language modeling approach to information retrieval. *Research and Development in Information Retrieval*, pages 275–281.

[Project,] Project, T. L. The Lemur toolkit for language modeling and information retrieval. http://www.lemurproject.org.

[Robertson and Walker, 1994] Robertson, S. E. and Walker, S. (1994). Some simple effective approximations to the 2-Poisson model for probabilistic weighted retrieval. In *SIGIR '94: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 232–241, New York, NY, USA. Springer-Verlag New York, Inc.

[Robertson et al., 1995] Robertson, S. E., Walker, S., Jones, S., Beaulieu, M. H., and Gatford, M. (1995). Okapi at TREC-3. In *Proceedings of TREC-3, the 3rd Text REtrieval Conference*, pages 109–127. NIST.

[Roelleke and Wang, 2006] Roelleke, T. and Wang, J. (2006). A parallel derivation of probabilistic information retrieval models. In *SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 107–114, New York, NY, USA. ACM.

[Schwarz, 1978] Schwarz, G. (1978). Estimating the dimension of a model. *Annals of Statistics*, 6(2):461–464.

[Zhai and Lafferty, 2002] Zhai, C. and Lafferty, J. (2002). Two-stage language models for information retrieval. In *SIGIR '02: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 49–56, New York, NY, USA. ACM.

[Zhai and Lafferty, 2004] Zhai, C. and Lafferty, J. (2004). A study of smoothing methods for language models applied to information retrieval. *ACM Transactions on Information Systems*, 2(2):179–214.

[Zhai and Lafferty, 2006] Zhai, C. and Lafferty, J. (2006). A risk minimization framework for information retrieval. *Inf. Process. Manage.*, 42(1):31–55.