An Investigation of Dirichlet Prior Smoothing's Performance Advantage

Mark D. Smucker[1] James Allan[1]

November 10, 2006

Abstract

In the language modeling approach to information retrieval, Dirichlet prior smoothing frequently outperforms Jelinek-Mercer smoothing. Both Dirichlet prior and Jelinek-Mercer are forms of linear interpolated smoothing. The only difference between them is that Dirichlet prior determines the amount of smoothing based on a document's length. Our hypothesis was that Dirichlet prior's performance advantage comes from an implicit document prior that favors longer documents. We tested our hypothesis by first calculating a prior for a given document length from the known relevant documents. We then determined the performance of each smoothing method with and without the document prior. We discovered that when given the document prior, Jelinek-Mercer smoothing matches or exceeds the performance of Dirichlet prior smoothing. Dirichlet prior smoothing's performance advantage appears to come more from an implicit prior favoring longer documents than from better estimation of the document model.

Keywords: Smoothing, Dirichlet prior, Jelinek-Mercer, language modeling, document prior.

1 Introduction

The language modeling approach to information retrieval represents documents as generative probabilistic models (Ponte and Croft 1998, Miller et al. 1998, Hiemstra and Kraaij 1998, Berger and Lafferty 1999, Song and Croft 1999). Documents with higher probabilities for query words are preferred over other documents. A document's score is computed to be the probability that it would generate the query. This probability is the product of each query word given the document's probabilistic model. The easiest way to estimate a model for a document is to assign a probability to each word appearing in the document equal to the number of times it occurs divided by the number of word occurrences in the document – this is known as maximum likelihood estimation. Words not in the document will be assigned a probability of zero. Zero probabilities are a problem; a document must contain all query words to avoid a score of zero.

To better estimate document models and eliminate zero probabilities, document models are *smoothed* to produce non-zero probabilities for all words. Common smoothing methods mix the document model with the collection model. The collection can be thought of as one large document consisting of all documents concatenated together. Mixing the document model with the collection model will produce a new document model that has some probability for all words. Query words not in the collection are dropped from the query. Smoothing techniques are commonly parameterized to control the amount of mixing between the document and collection model.

Zhai and Lafferty (2001) investigated the use of three types of smoothing in information retrieval. They reported on Jelinek-Mercer, Dirichlet prior, and absolute discounting smoothing methods. They looked at the performance attainable by these methods on nine collections using both short and very long queries. They used the TREC topic's keyword-like title field for short queries and a concatenation of the title, description, and narrative fields for the long queries. Jelinek-Mercer and Dirichlet prior were the better performing methods. On the short queries, Dirichlet prior smoothing was the best performing on eight of the nine collections with absolute discounting being the best on one collection. The performance difference on the short queries was large with an average mean average precision (MAP) of 0.256 for Dirichlet prior vs. 0.227 for Jelinek-Mercer across the nine collections. On the long queries, Dirichlet prior (DP) was the best on six collections and Jelinek-Mercer (JM) smoothing was best on the other three, but their average performance was essentially equivalent. On the long queries, the average MAP for DP was 0.279 vs. 0.280 for JM. In a later work, Zhai and Lafferty reported on JM vs. DP performance when the sentence-length description field was used as a query (Zhai and Lafferty 2002). Out of six collections, DP performed better than JM on five with an average MAP of 0.211 compared to JM's average MAP of 0.187. The title and description queries represent query lengths one could realistically expect from a user, and on these lengths Dirichlet prior considerably outperforms Jelinek-Mercer smoothing.

In this paper, we attempt to answer why the Dirichlet prior (DP) performs better than Jelinek-Mercer (JM) smoothing. As we explain later, both DP and JM smooth identically except that DP determines the amount of smoothing based on a document's length. Our hypothesis is that DP has an implicit prior that prefers longer documents, which is advantageous on the TREC collections. To test our hypothesis, we calculate for each set of queries the probability of relevance given the document length and use this as a document prior. We then determine the performance attainable by these two smoothing methods given the document priors. As we will show, when JM is given a document prior based on length, its performance equals or betters that of the Dirichlet prior both with and without the prior. Dirichlet prior smoothing is unable to leverage the document prior and in some cases is even hurt by the prior, which suggests that the given document prior conflicts with Dirichlet prior smoothing's implicit document prior.

2 Methods and Materials

2.1 Notation

The vocabulary, V, is the set of words in the collection. The number of words in V is $|V|$. Documents are multisets (bags) over V. A document, D, is a function $D : V \rightarrow \mathbb{N}$ where $\mathbb{N} = \{0, 1, 2, \ldots\}$ is the set of natural numbers. The multiplicity of w in D, $D(w)$, is the count of word w in document D. The document length is the cardinality of D, $|D|$ and is defined as follows:

$$|D| = \sum_{w \in V} D(w)$$

The collection, C, is also a multiset over V and $|C|$ is the total number of word occurrences in C. Query Q is also represented as a multiset over the vocabulary.

The probabilistic model of document D will be represented as M_D. The probability of a word w given a document model M_D is $P(w|M_D)$. For convenience, we write the maximum likelihood estimated (MLE) probability of a word w given a piece of text T as $P(w|T)$.

2.2 Probabilistic Models of Documents

We use the multinomial as our probabilistic model of text. A multinomial model of text specifies a probability for each word in the vocabulary V. The probabilities of the multinomial are its parameters and thus there are $|V|$ parameters. The probabilities of the multinomial sum to 1. A common way to think about the multinomial is as a biased die. A die has $|V|$ faces with each word having some probability of being *generated* by the die on a roll.

For a given text document, D, the parameters of the multinomial representing D, M_D, need to determined. This process of computing the probability of a word w given the model M_D, $P(w|M_D)$, is called estimation. A standard approach to parameter estimation is maximum likelihood estimation (MLE). MLE maximizes the likelihood of the observed data given the model. Treating the words of D as independent samples, the likelihood of D is defined to be:

$$L(D) = \prod_{w \in D} P(w|M_D)^{D(w)} \tag{1}$$

The maximum likelihood estimate for the probability of a word turns out to be the count of that word divided by the total number of occurrences in D:

$$P(w|D) = P(w|M_D) = \frac{D(w)}{|D|} \tag{2}$$

We can create MLE models of any piece of text T. As mentioned in the previous section, we will write $P(w|T)$ to represent the MLE probability of w given T.

Note that the MLE model has zero probabilities for all words not in the document.

2.3 Retrieval Model

Documents are ranked by the probability of a document given a query, which is given by Bayes' theorem as:

$$P(D|Q) = \frac{P(Q|D)P(D)}{P(Q)} \tag{3}$$

We drop $P(Q)$ from the above equation since it is the same for all documents and will not affect the ranking. The prior probability of a document is given by $P(D)$.

The probability that a document model could generate a query Q, which is known as query likelihood, is given by:

$$P(Q|M_D) = \prod_{w \in Q} P(w|M_D)^{Q(w)} \tag{4}$$

where M_D is the probabilistic model of D. Plugging Equation 4 into Equation 3 (with $P(Q)$ dropped) gives us our scoring function for a document:

$$P(D|Q) = P(D) \prod_{w \in Q} P(w|M_D)^{Q(w)} \tag{5}$$

If for some word $w \in Q$, $P(w|M_D) = 0$, then the document will be given a score of zero. This is the zero probability problem. Clearly we cannot directly use the MLE model of document. We can eliminate the zero probabilities from the document models using smoothing.

2.4 Document Smoothing Methods

A solution to the problem of zero probabilities and poor probability estimates is to bring prior knowledge to the estimation process. A natural fit as a prior for the multinomial is the Dirichlet density (Sjölander et al. 1996). A Dirichlet density can be thought of as an urn containing multinomial dies. All the multinomials are of the same size with $|V|$ parameters. The Dirichlet density has the same number of parameters as the multinomials for which it is a prior. The vector $\vec{\alpha}$ represents the parameters of the Dirichlet density. For each word w in the vocabulary, there is a corresponding element α_w of $\vec{\alpha}$, and all $\alpha_w > 0$.

When we use the Dirichlet density as the prior for the multinomial, the estimate of the probability of word given a document is the weighted average of the word's probability in all multinomials. Each multinomial is weighted by its probability given the observed document and the Dirichlet density. This estimate is the mean posterior estimate:

$$P(w|M_D) = \int_M P(w|M) P(M|\vec{\alpha}, D) dM \tag{6}$$

which reduces to:

$$P(w|M_D) = \frac{D(w) + \alpha_w}{|D| + |\vec{\alpha}|} \tag{7}$$

as shown by Sjölander et al. (1996).

The longer the document, the less influence the Dirichlet prior has in determining the parameter estimates for the multinomial M_D. The mean of the Dirichlet density is for each α_w, $\alpha_w/|\vec{\alpha}|$. As the text becomes shorter, the parameter estimates for M_D regress to the mean of the Dirichlet density.

The parameters of a Dirichlet density can be represented as a multinomial probability distribution M and a weight $m = |\vec{\alpha}|$. Thus, with $P(w|M) = \alpha_w/|\vec{\alpha}|$, Equation 7 becomes:

$$P(w|M_D) = \frac{D(w) + mP(w|M)}{|D| + m} \tag{8}$$

The machine learning community terms this formulation of Dirichlet prior smoothing the *m-estimate* (Mitchell 1997). The parameter m is the *equivalent sample size.* The Dirichlet density when used as a prior for the multinomial can be understood as taking m samples according to $P(w|M)$ prior to observing the data in D.

The parameters of the Dirichlet density can be determined using maximum likelihood estimation (MLE). MLE finds the density parameters that produce the highest likelihood for a collection of documents when the density is used as a prior. The MLE can be computed numerically using a Newton-Raphson method (Narayanan 1991) or via an expectation maximization (EM) like method (Sjölander et al. 1996).

In contrast, common practice in information retrieval, and the one we follow, is to let $P(w|M) = P(w|C)$, i.e. use the MLE model of the collection for M. This results in the common expression of Dirichlet prior smoothing as:

$$P(w|M_D) = \frac{D(w) + mP(w|C)}{|D| + m} \tag{9}$$

The value of m is a fixed value and is determined empirically. Typically m is set to maximize a retrieval metric like mean average precision for a set of queries and a collection of documents.

A closely related smoothing method is *linear interpolated smoothing.* Linear interpolated smoothing linearly combines two models to produce a smoothed model. Documents are typically smoothed with the collection. The document D is smoothed with the collection C as follows:

$$P(w|M_D) = (1 - \lambda)P(w|D) + \lambda P(w|C) \tag{10}$$

The amount of smoothing increases as λ increases from 0 to 1, which matches the behavior of an increase in Dirichlet prior's parameter m. $P(w|D)$ and $P(w|C)$ are the MLE models of D and C respectively.

Dirichlet prior smoothing is a form of linear interpolated smoothing. For a given document length $|D|$ and parameter m, an equivalent λ exists. By setting λ in Equation 10 to:

$$\lambda = 1 - \frac{|D|}{|D| + m} \tag{11}$$

Equation 9 can be written in the form of Equation 10 (Johnson 1932):

$$\begin{aligned} P(w|M_D) &= (1 - 1 + \frac{|D|}{|D| + m})P(w|D) + (1 - \frac{|D|}{|D| + m})P(w|C) \\ &= \frac{P(w|D)|D|}{|D| + m} + (\frac{|D| + m}{|D| + m} - \frac{|D|}{|D| + m})P(w|C) \\ &= \frac{D(w)}{|D| + m} + \frac{mP(w|C)}{|D| + m} \\ &= \frac{D(w) + mP(w|C)}{|D| + m} \end{aligned}$$

Because Dirichlet prior is equivalent to linear interpolated smoothing with a λ parameterized on document length, both methods smooth a document exactly the same way for a given λ.

As document length increases, Dirichlet prior smoothing gives less weight to the collection, $P(w|C)$, and more weight to the document, $P(w|D)$. Longer documents' maximum likelihood estimates of probabilities are trusted more than shorter documents' estimates.

We studied the difference between Dirichlet prior smoothing and linear interpolated smoothing with a fixed λ. Following Chen and Goodman (1998) and Zhai and Lafferty (2001), we will refer to linear interpolated smoothing with a fixed λ as Jelinek-Mercer smoothing.

2.5 Hypothesis

The only difference between Dirichlet prior and Jelinek-Mercer smoothing is that Dirichlet prior determines the amount of smoothing based on a document's length as per Equation 11. As such, the performance gains obtained by Dirichlet prior are a function of document length. Singhal et al. (1996) have reported that in the TREC collections longer documents are more likely to be relevant. When a document is smoothed using linear interpolated smoothing, the probability estimates for the words in the document are moved closer to the probabilities of the words in the collection. For informative words in relevant documents, the collection probability is likely to be much smaller than the document probability. Thus, at least for documents containing all the query terms, the more a document is smoothed, the lower that document will score. Given that Dirichlet prior smoothing smooths shorter documents more than longer documents, we hypothesize that Dirichlet prior smoothing's improved performance comes from favoring longer documents over shorter documents. In other words, we hypothesize that Dirichlet prior smoothing has an *implicit* document prior, $P(D)$ in Equation 5, which assigns higher prior probabilities of relevance to longer documents.

One way to test our hypothesis would be to transform a TREC collection into a collection with a uniform probability of relevance given document length. Such a transformation would consist of deleting both relevant and non-relevant documents of various lengths. We rejected this option of transforming the collection because a smaller collection size is undesirable and fewer relevance judgments would compromise our ability to evaluate retrieval results accurately.

The retrieval method of section 2.3 provides us with another means to test our hypothesis. We can compute the known prior probability of relevance given document length using a set of queries' relevance judgments. We can then directly supply this prior probability of relevance as the prior, $P(D)$, in Equation 5. A calculation of $P(Q|D)$ unbiased by document length should be correctly adjusted by our document prior to produce a superior $P(D|Q)$ ranking. If a smoothing method produces a length biased $P(Q|D)$, supplying our document prior will interfere and may help or hurt ranking depending on the method's implicit prior probability of relevance given document length.

Neither Jelinek-Mercer (JM) nor Dirichlet prior (DP) smoothing are supposed to incorporate any preference for documents based on length. A smoothing method's sole responsibility is to better estimate a document's true probabilistic model. As such, both JM and DP should show performance improvements given this document prior unless they already include an implicit document prior based on length.

Our use of the document prior is inspired by the technique of pivoted length normalization (Singhal et al. 1996) but is different in form and intent. Singhal et al. (1996) created a length normalization process so that the distribution of retrieved document lengths would match the distribution of relevant document lengths. If our goal was to correct for retrieval length biases, our document prior would be computed differently. We would adjust the prior until the distribution of retrieved lengths matched that of the relevant documents. Instead, we supplied a true document prior based on length to JM and DP smoothing and measured the effect on retrieval performance.

2.6 Document Priors

For each set of experimental topics, we calculated document priors for a given document length. The document prior is computed as the number of relevant documents at a given length divided by the total number of documents at that length. If a given length had less than 1000 documents, we created a bin and grew it to cover greater lengths until it contained at least 1000 documents. (The bin containing the longest documents, was smaller than 1000 documents. These bins had sizes of 957, 636, and 577 documents for TREC 3, 7, and 8 respectively.) A bin's length is the average of the document lengths in the bin. We then smoothed the probabilities using the lowess smoother built into the R statistical package with its delta equal to ten (R Development Core Team 2004). The resulting curve was used to determine the prior probability of a document based on its length using interpolation. If during retrieval a document was found outside the range of the smoothed curve, the document was given the same prior as the nearest bin. For TREC 3, the lowess curve went negative for very short documents, and we set their probability to 1e-6. Figures 1, 2, and 3 show the computed probabilities for each bin and the smoothed curves.

[Figure 1 about here.]

[Figure 2 about here.]

[Figure 3 about here.]

Figure 4 shows how TREC 3 has a very strong bias for relevant documents to be between 500 and 1000 terms in length. Documents shorter than 250 terms are very unlikely to be relevant relative to the other document lengths. TREC 7 and 8 are not nearly as biased, but for them also, as documents become longer, they are more likely to be relevant.

[Figure 4 about here.]

2.7 Experiments

The experiments consisted of determining the performance of Jelinek-Mercer and Dirichlet prior smoothing with and without the document prior based on document length.

If we were testing the document prior as a method to improve retrieval performance, then we would use the document priors from one set of topics as a training set and test on another set of topics. Instead, we purposely use the document priors calculated for a set of topics with that set of topics. By providing the correct document priors, we are able to eliminate any implicit length preference as a means to better performance. With the given document priors, any performance advantage that a smoothing method shows must come from better estimation of document models or other unknown features of the smoothing method.

We used the TREC 3, 7, and 8 ad-hoc retrieval tasks for our experiments. These tasks respectively consist of topics 151-200, 351-400, and 401-450. Each topic consists of a title, description, and narrative. The titles best approximate a short keyword query while the description is typically formulated as a single well formed sentence describing the information need of the user. The narratives are directions to potential future relevance assessors and are often paragraph length descriptions of what should be considered on-topic and off-topic.

We used only titles and descriptions in isolation of each other to represent queries. For Zhai and Lafferty (2001), the short queries are the titles and the long queries are the concatenation of title, description and narrative fields. We agree with the formulation to use titles as keyword-like non-verbose queries and descriptions as verbose queries (Zhai and Lafferty 2002). A verbose query is likely to contain many more common and non-informative words as opposed to the more focused title queries.

The collection for TREC 3 consists of TREC volumes (discs) 1 and 2. The collection for TREC 7 and 8 consists of TREC volumes 4 and 5 minus the Congressional Record (CR) subcollection. We preprocessed the collections and queries in the same manner. We stemmed using the Krovetz stemmer (Krovetz 1993) and removed stopwords using an in-house stopword list of 418 noise words. We used Lemur 2.0.3 (Lemur 2003) compiled to run on Windows XP for all experiments. Documents were scored using query likelihood, and we modified Lemur to use document priors.

The parameters for Dirichlet prior and Jelinek-Mercer smoothing were determined by evaluating the mean average precision for a set of parameter values. For Dirichlet prior, m was tried with values of $\{50, 100, 150, 200, 250, 300, 350, 400, 500, 600, 800, 1000, 1250, 1500, 1750, 2000, 2500, 3000, 5000\}$. For Jelinek-Mercer, λ was tried with values of 0.05 to 0.95 inclusive in 0.05 increments.

All statistical significance results are with respect to a paired, two sided, Student's t-test, and if not stated, $p < 0.05$.

3 Results

Table 1 summarizes the results of the experiments. Without the document prior, Dirichlet prior (DP) smoothing significantly outperforms Jelinek-Mercer (JM) smoothing on every set of topics and query type. These results reproduce those of Zhai and Lafferty (2001, 2002) that also show the performance advantage of DP smoothing. In each case without the document prior, the difference between DP and JM is statistically significant at a $p < 0.05$ level by paired, two sided, Student's t-test.

[Table 1 about here.]

JM smoothing with the document prior matches or exceeds the performance of DP smoothing both with and without the document prior in all cases. On the description queries, JM with a prior has a better performance than DP with a prior at a statistically significant level ($p < 0.04$) on topics 151-200 and 401-450 and is equivalent to DP on topics 351-400. In all cases, the performance of JM is significantly increased by the addition of the prior and some increases are dramatic. For example, on description queries for topics 151-200, JM's mean average precision increases 23.5% from 0.183 to 0.226. The performance of DP smoothing with the document prior is equal to or worse than DP smoothing without the prior. For all description queries, DP smoothing is hurt by the use of the document prior.

Table 2 shows that DP used less smoothing with the document prior. In contrast, JM used the same or an increased amount of smoothing. Compared to DP, JM uses considerably more smoothing for the majority of documents in the collection. Both methods used more smoothing for description queries than for the title queries.

[Table 2 about here.]

4 Discussion

When Jelinek-Mercer (JM) smoothing is given the document prior, it matches or exceeds the performance of Dirichlet prior (DP) smoothing with and without the prior. When DP smoothing is given the document prior, its performance stays the same or worsens. In effect, DP's implicit document prior interferes with the given document prior based on length. To compensate for the given document prior, DP uses less smoothing than when used without the document prior. As JM smoothing shows though, a large amount of smoothing can be used for better performance with the given document prior.

These experimental results offer strong support for our hypothesis. Dirichlet prior's performance advantage over Jelinek-Mercer appears to come more from an implicit prior favoring longer documents than from better estimation of the document model.

These results lead directly to two questions. First, is it correct to smooth longer documents less as Dirichlet prior smoothing does? We will next show that smoothing longer documents less appears to be correct from an estimation standpoint. Second, why do both smoothing methods need to use a large amount of smoothing to obtain good retrieval performance? Dirichlet prior uses more smoothing than appears to be required for good estimation, and Jelinek-Mercer is successful with large amounts of smoothing for all document lengths. We will explain that because the amount of smoothing controls the amount of smoothing's inverse document frequency (IDF) like behavior (Zhai and Lafferty 2001), one needs significant amounts of smoothing to maximize that behavior.

Finally, we discuss the potential of using a document prior for performance improvements.

4.1 Smoothing Longer Documents Less

Outside of the advantage of preferring longer documents, does it makes sense to smooth longer documents less? Linear interpolated smoothing (and thus Dirichlet prior) is a discounting smoothing method. Discounting methods reduce the probability of the words seen and reallocate the probability mass to words not seen in the document. The mass assigned to the unseen words is called the *zero probability mass.* Neither Jelinek-Mercer nor Dirichlet prior smoothing specify the amount of discounting explicitly but instead an increase in the value of their smoothing parameters results in more discounting. Good-Turing is another form of discounted smoothing. Good-Turing explicitly uses the zero probability mass, P_0, and estimates it for a document D to be:

$$P_0 = \frac{\mathrm{N}_1(D)}{|D|}$$

where $\mathrm{N}_1(D)$ is the number of words that occur exactly once in the document D (Sampson 2001). We will not use or discuss Good-Turing smoothing beyond using its estimation of the zero probability mass. Gale and Sampson (1995) provide a good explanation of Good-Turing smoothing.

The λ parameter for linear interpolated smoothing can be determined directly from the Good-Turing estimate of the zero probability mass. To do this, we take the sum of the seen probabilities and set the sum equal to $1 - P_0$ and solve for the smoothing parameter. For linear interpolated smoothing, the P_0 derived λ is:

$$\begin{aligned} \sum_{w \in D} ((1-\lambda)P(w|D) + \lambda P(w|C)) &= 1 - P_0 \\ \lambda &= \frac{P_0}{1 - \sum_{w \in D} P(w|C)} \end{aligned} \tag{12}$$

A similar derivation can be done for the Dirichlet prior parameter m, but this merely produces an identical smoothing method. Using Equation 12, we can determine the amount to smooth each separate document based on the Good-Turing estimate of its zero probability mass.

[Figure 5 about here.]

Figure 5 shows the P_0 derived λ values for a random set of two thousand documents from the 1.6 million documents comprising TREC disks 1-5 minus the CR collection on discs 4 and 5. The curve marked “Average” is the average of the 1.6 million documents' P_0 derived λ's after binning the documents by length. To produce a smoother average curve, each bin has a minimum of 1000 documents and at least 2 document lengths. Also shown are the equivalent λ values for two settings of the Dirichlet prior parameter m at 1000 and 320. Dirichlet prior follows the general trend of the P_0 derived λ values; longer documents receive less smoothing than shorter documents. Jelinek-Mercer (JM) smoothing, on the other hand, smooths long and short documents equally and is seen as a horizontal line in the figure for $\lambda = 0.8$. It thus could be argued that Dirichlet prior is correct in smoothing longer documents less if we believe in the Good-Turing estimate of the zero probability mass, P_0. In comparison, JM smoothing appears to use too little smoothing for very short documents and smooths long documents too much.

Zhai and Lafferty (2002) created a "leave one out" method to estimate the Dirichlet prior parameter m. Their method finds the m that minimizes the log likelihood for a modified collection where each document's likelihood is computed by removing a word and smoothing the resulting document with Dirichlet prior smoothing. This method is part of the Lemur software distribution (Lemur 2003), and we used it to calculate m for the TREC volumes 1 and 2 collection used for TREC 3 and the TREC volumes 4 and 5 collection, minus the CR subcollection, used for TREC 7 and 8. For volumes 1 and 2, the estimate of m is 308 and for volumes 4 and 5 minus CR the estimate of m is 332. The average of 308 and 332 is 320 and as can be seen in Figure 5 appears to be a reasonable fit to the P_0 derived λ values. Thus both Zhai and Lafferty's method and the Good-Turing estimate of the zero probability mass appear to be in agreement with each other. Both of these methods call for much less smoothing than is needed for good document retrieval. We next discuss why so much smoothing is used by both Dirichlet prior and Jelinek-Mercer.

4.2 IDF Behavior of Smoothing

If it is correct to use as little smoothing as suggested by the Good-Turing estimate of the zero probability mass and Zhai and Lafferty's leave-one-out estimates, then why do Dirichlet prior and Jelinek-Mercer use so much more smoothing than appears to be needed for good estimation? Zhai and Lafferty (2001) have shown that smoothing the document model with the collection model can be viewed as introducing an inverse document frequency (IDF) like behavior to the query likelihood retrieval model. Their and our experimental results show that longer, verbose queries require more document smoothing than shorter queries. As we will illustrate with an example, high levels of smoothing increase the importance of rare terms relative to common terms. In other words, the IDF-like behavior shown to exist by Zhai and Lafferty is accentuated with high levels of smoothing.

When a document is scored using query likelihood as in Equation 4, each term in the query contributes to the document's score. When ranking documents, it is their scores relative to each other that matters. If a query consisted of two words w_1 and w_2, the ratio of a document A to a document B tells us to what extent either one is more likely to have generated the query:

$$\frac{P(w_1|M_A)P(w_2|M_A)}{P(w_1|M_B)P(w_2|M_B)}$$

where M_A and M_B are the smoothed models of documents A and B. This ratio is simply a product of the ratios for each word. The ratio for word w_1 is:

$$\frac{(1-\lambda)P(w_1|A)+\lambda P(w_1|C)}{(1-\lambda)P(w_1|B)+\lambda P(w_1|C)}$$

Let $P(w_1|A) = P(w_2|A) = 0.003$ and $P(w_1|B) = P(w_2|B) = 0.001$. Document A is the superior document. When $\lambda = 0$ the ratio of A's score to B's is 9:1 and each word contributes equally to A's higher score. If we increase λ, the individual word ratios will change from 3:1 to ratios nearer to 1:1 until $\lambda = 1$ and the 1:1 ratio is obtained. The way the individual ratios change though depends on their respective collection probabilities.

Let us further assume that w_1 is a rare term and w_2 is a common term. To determine what makes a term rare or common, we can look at the actual collection probabilities for words found in the description queries. The words used in description queries are skewed to rare informative words, but many common and less-informative words are also used. For topics 351-450, the minimum collection probability of a query term is 7.3×10^{-8} and the maximum is 3.1×10^{-3}. The median probability is 2.6×10^{-4}. Let us assume that the first quartile is a good representative of a rare term and the third quartile represents a common word. We thus let $P(w_1|C) = 6.0 \times 10^{-5}$ (rare) and $P(w_2|C) = 4.6 \times 10^{-4}$ (common).

[Figure 6 about here.]

Figure 6 shows the scenario just described. As λ increases from 0 to 1, the 3:1 ratio for each word changes at different rates. The rare word w_1 has a document probability that is large relative to its collection probability and thus requires significantly more smoothing to affect the ratio between documents A and B. The common word being closer to its collection probability moves faster to a 1:1 ratio as smoothing is increased. For this example, the result is that at $\lambda = 0.86$ the effective power of the rare word over the common word is maximized.

Informative words are characterized as occurring in bursts and being unevenly distributed in the collection while non-informative words are more evenly distributed (Church 2000). A common heuristic to identify informative words is the inverse document frequency (IDF). In the language modeling approach with documents smoothed with the collection, IDF is replaced by the inverse collection probability, which functions similarly. Informative, rare words will tend to have large document probabilities relative to the collection probability. Thus for informative words their influence on a document's ranking is little changed until large amounts of smoothing are applied. Common words will likely have document probabilities already near the collection probabilities. Thus common words lose their influence on a document's ranking much faster than rare words as the amount of smoothing increases.

The power of rare words will tend to be amplified with high levels of smoothing. This is the likely explanation of why DP and JM smoothing succeed with so much smoothing even when for estimation purposes they should be using less smoothing. This is a surprising notion given that increased smoothing should be used to correct poor model estimates. Instead we find that smoothing more, but not too much, increases the weight given to rarer words in a query. In other work, we've found that document retrieval performance can increase if we decouple smoothing's IDF and estimation roles (Smucker and Allan 2006).

4.3 Document Priors for Improved Performance

While not the focus of this paper, an obvious question to ask is whether or not the document priors as calculated can be used to improve retrieval performance. While a full analysis is beyond the scope of this paper, we looked at how sensitive performance is to the document prior. As Figures 1-4 show, the priors for topics 351-400 and 401-450 are similar while the priors for topics 151-200 are quite different. Recall that topics 351-400 and 401-450 use the same collection. These curves imply that priors for a collection do not vary much based on topics but that priors for one collection may be quite different from another.

[Table 3 about here.]

To examine the sensitivity, we used the document priors from each set of topics with the other set of topics. We only look at Jelinek-Mercer (JM) smoothing since Dirichlet prior smoothing did not show a performance improvement with the document priors. Table 3 shows the mean average precision (MAP) for each set of topics and queries using the different document priors. The MAP scores were found with the same parameter sweep of λ as described in Section 2.7.

Not surprisingly, swapping the document priors for topics 351-400 and 401-450 leads to similar performance. Interestingly, equivalent performance is obtainable by using the document priors computed for topics 151-200 on topics 351-400 and 401-450. When the priors for topics 351-400 and 401-450 are used on topics 151-200, performance degrades, but the performance is still greater than that without a prior.

It appears as though the use of a document prior based on length computed from different topics and collections may lead to performance improvements on other topics and collections. In particular, the success of the prior for topics 151-200 on the other topics suggests that the most important aspect of the document priors may be to penalize very short documents.

5 Conclusion

We discovered that Dirichlet prior's performance advantage over Jelinek-Mercer smoothing comes more from an implicit document prior that prefers longer documents than from an ability to better estimate the true document model. We determined this by constructing a prior for a given document length from the known relevant documents. We then tested the performance of Dirichlet prior and Jelinek-Mercer smoothing with and without the document prior. Both methods smooth documents identically except Dirichlet prior smooths longer documents less. With the prior, Jelinek-Mercer smoothing equals or betters the performance of Dirichlet prior smoothing. By smoothing longer documents less, Dirichlet prior smoothing favors them. Smoothing longer documents less does make sense from an estimation standpoint, but Jelinek-Mercer smoothing's better performance on description queries seems to occur because at higher levels of smoothing, the IDF-like behavior of document smoothing appears to be maximized. Finally, we found that a document prior based on another collection has the potential for improved performance. In particular, it appears that one should penalize the scores of very short documents.

Acknowledgments

This work was supported in part by the Center for Intelligent Information Retrieval, the Defense Advanced Research Projects Agency (DARPA) under contract number HR0011-06-C-0023, and by SPAWARSYSCEN-SD grant number N66001-02-1-8903. Any opinions, findings and conclusions or recommendations expressed in this material are the authors' and do not necessarily reflect those of the sponsor.

References

Berger A and Lafferty J (1999) Information retrieval as statistical translation. In: SIGIR '99: Proceedings of the 22th annual international ACM SIGIR conference on research and development in information retrieval. ACM Press, pp. 222–229.

Chen SF and Goodman J (1998) An empirical study of smoothing techniques for language modeling. Tech. Rep. TR-10-98, Center for Research in Computing Technology, Harvard University.

Church KW (2000) Empirical estimates of adaptation: the chance of two Noriegas is closer to $p/2$ than p^2. In: Proceedings of the 17th Conference on Computational Linguistics. Association for Computational Linguistics, pp. 180–186.

Gale WA and Sampson G (1995) Good-Turing frequency estimation without tears. Journal of Quantitative Linguistics, 2(3):217–237. Reprinted (Sampson 2001, chap. 7).

Hiemstra D and Kraaij W (1998) Twenty-One at TREC-7: Ad-hoc and cross-language track. In: The Seventh Text REtrieval Conference (TREC-7). Department of Commerce, National Institute of Standards and Technology, pp. 227–238.

Johnson WE (1932) Probability: deductive and inductive problems. Mind, 41(164):409–423.

Krovetz R (1993) Viewing morphology as an inference process. In: SIGIR '93: Proceedings of the 16th annual international ACM SIGIR conference on research and development in information retrieval. ACM Press, pp. 191–202.

Lemur (2003) Lemur Toolkit for Language Modeling and IR. `http://www.lemurproject.org/`.

Miller DRH, Leek T and Schwartz RM (1998) BBN at TREC7: Using hidden markov models for information retrieval. In: The Seventh Text REtrieval

Conference (TREC-7). Department of Commerce, National Institute of Standards and Technology, pp. 133–142.

Mitchell TM (1997) Machine Learning. McGraw-Hill.

Narayanan A (1991) Algorithm as 266: Maximum likelihood estimation of the parameters of the dirichlet distribution. Applied Statistics, 40(2):365–374.

Ponte JM and Croft WB (1998) A language modeling approach to information retrieval. In: SIGIR '98: Proceedings of the 21th annual international ACM SIGIR conference on research and development in information retrieval. ACM Press, pp. 275–281.

R Development Core Team (2004) R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. 3-900051-07-0.

Sampson G (2001) Empirical Linguistics. Continuum.

Singhal A, Buckley C and Mitra M (1996) Pivoted document length normalization. In: SIGIR '96: Proceedings of the 19th annual international ACM SIGIR conference on research and development in information retrieval. ACM Press, pp. 21–29.

Sjölander K, Karplus K, Brown M, Hughey R, Krogh A, Mian IS and Haussler D (1996) Dirichlet mixtures: a method for improved detection of weak but significant protein sequence homology. Computer Applications in the Biosciences, 12:327–345.

Smucker MD and Allan J (2006) Lightening the load of document smoothing for better language modeling retrieval. In: SIGIR '06: Proceedings of the 29th annual international ACM SIGIR conference on research and development in information retrieval. ACM Press, pp. 699–700.

Song F and Croft WB (1999) A general language model for information retrieval. In: CIKM '99: Proceedings of the eighth international conference on information and knowledge management. ACM Press, pp. 316–321.

Zhai C and Lafferty J (2001) A study of smoothing methods for language models applied to ad hoc information retrieval. In: SIGIR '01: Proceedings of the 24th annual international ACM SIGIR conference on research and development in information retrieval. ACM Press, pp. 334–342.

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. ACM Press, pp. 49–56.

Notes

[1] Center for Intelligent Information Retrieval, Department of Computer Science, University of Massachusetts, Amherst, MA 01003. Email: smucker@cs.umass.edu and allan@cs.umass.edu

List of Figures

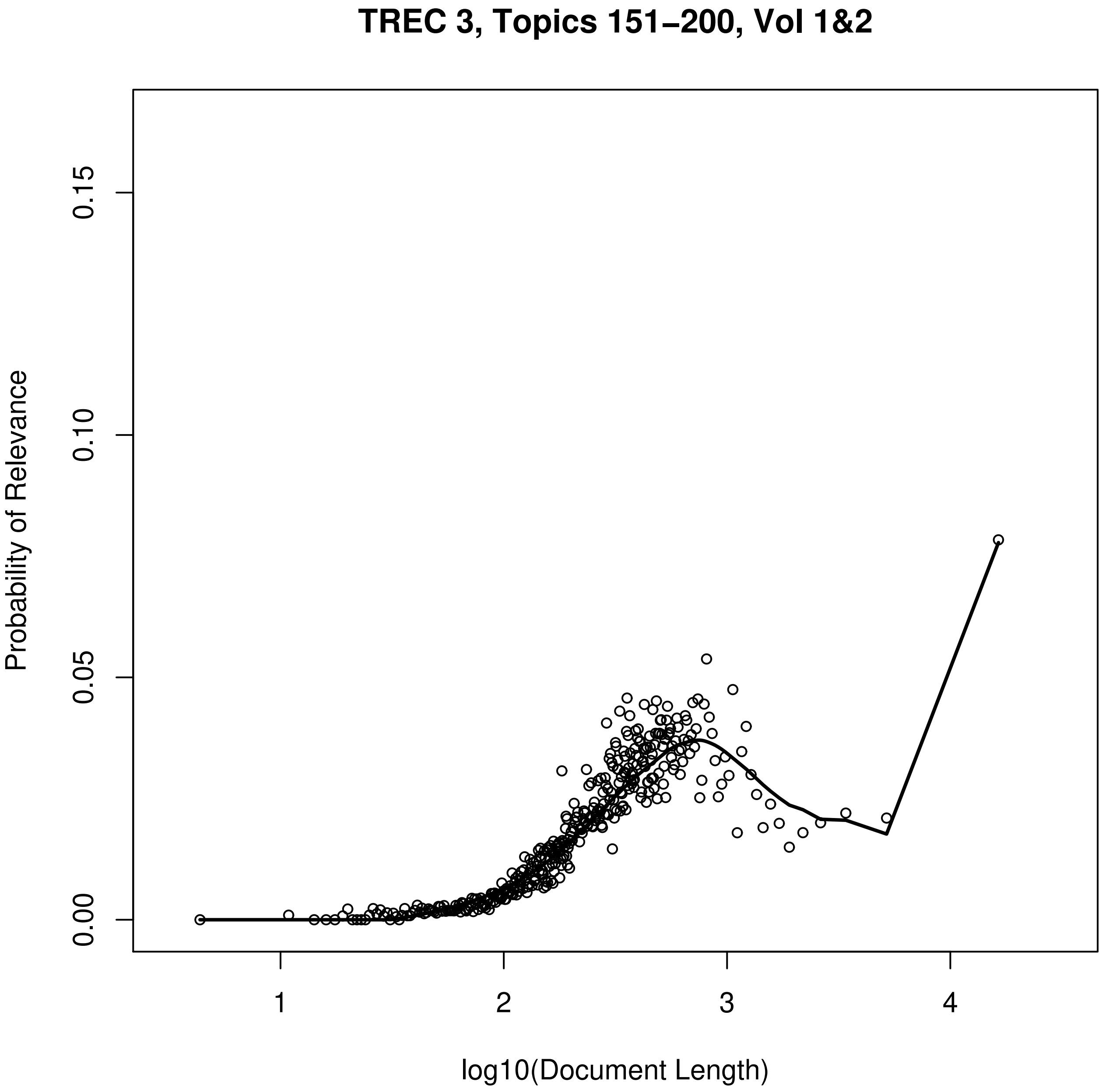

Figure 1: The computed probability of relevance for documents binned by length for TREC 3, topics 150-200. The curve through the points represents the smoothed probabilities used for retrieval.

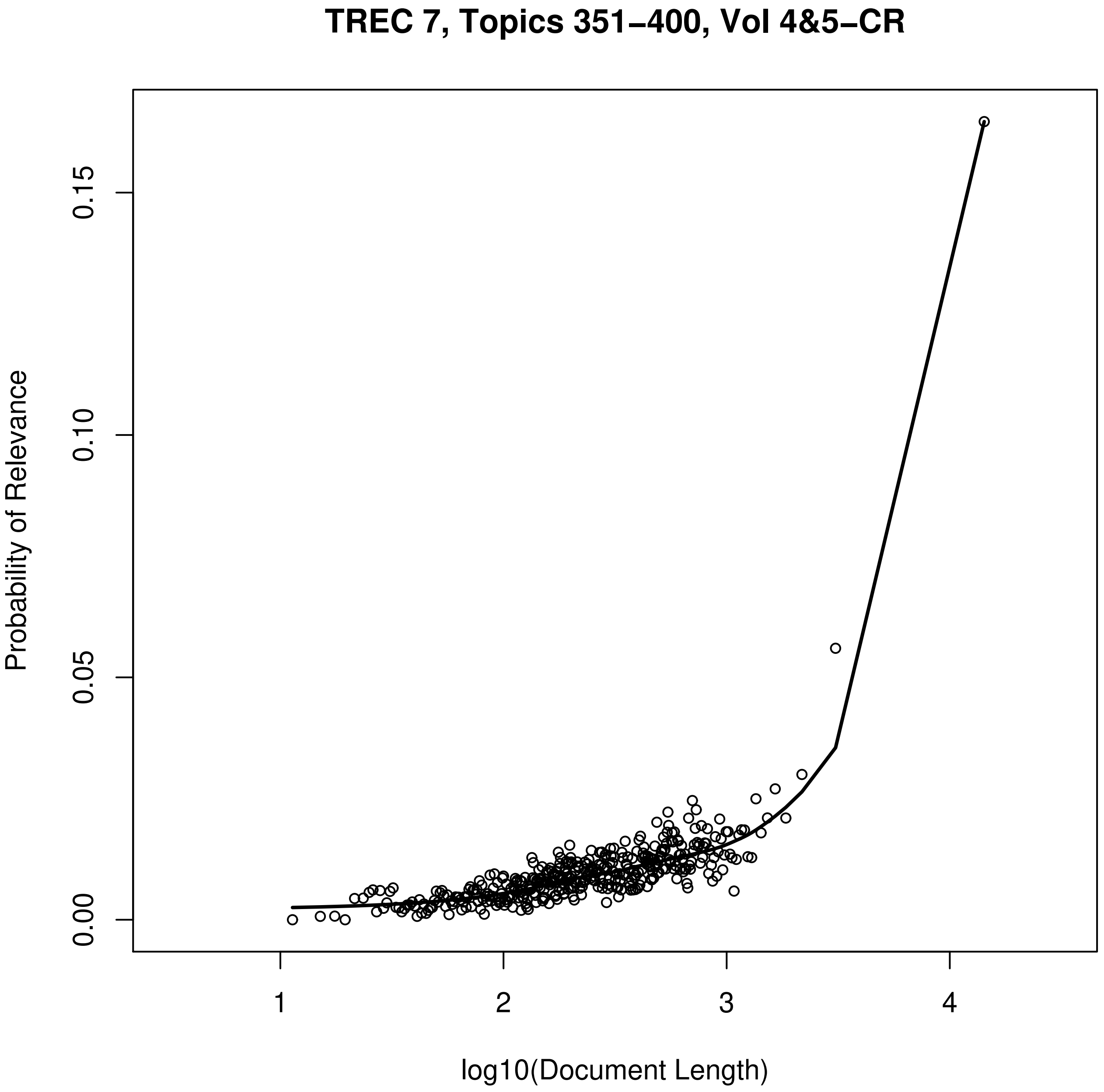

Figure 2: The computed probability of relevance for documents binned by length for TREC 7, topics 350-400. The curve through the points represents the smoothed probabilities used for retrieval.

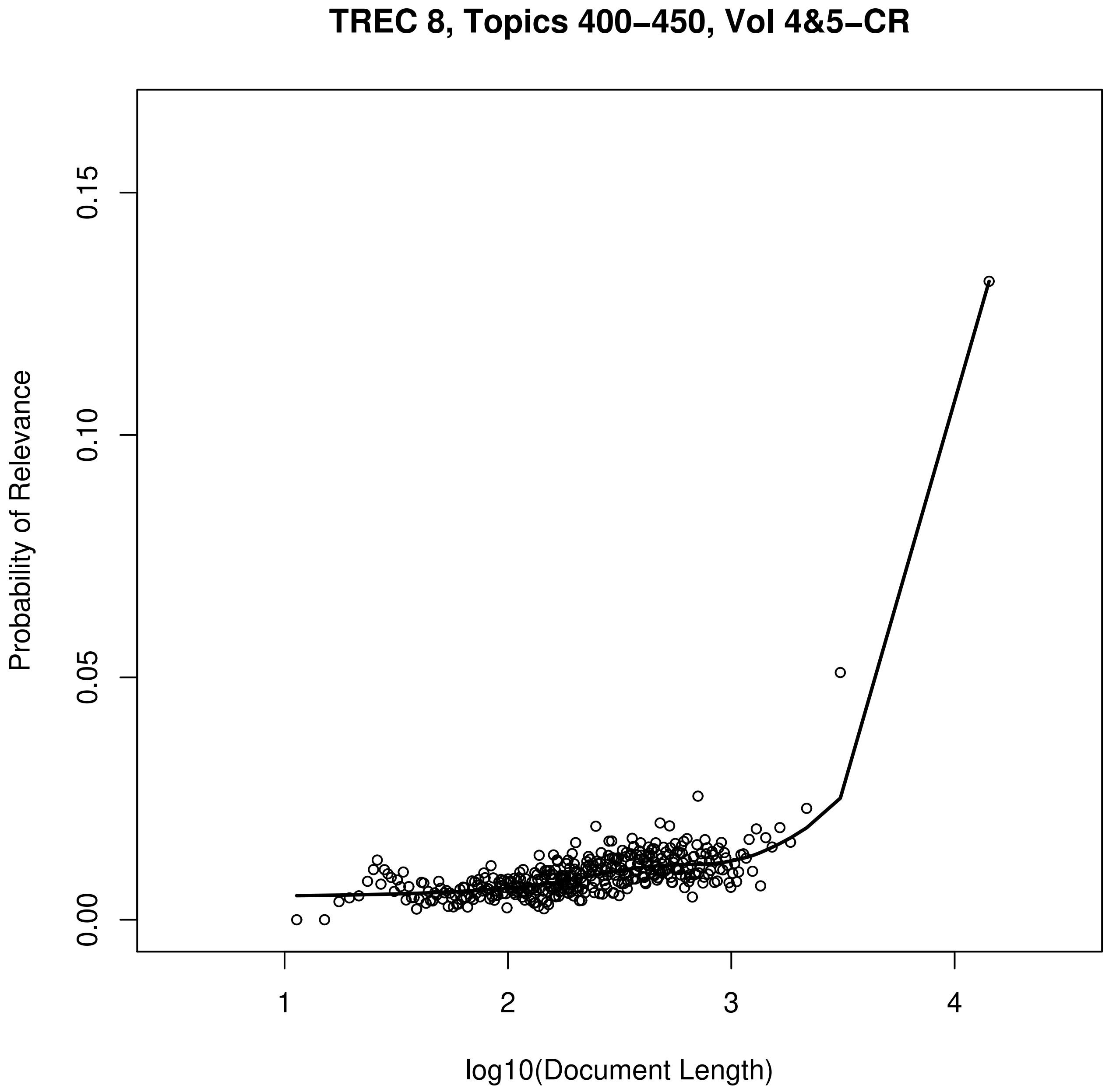

Figure 3: The computed probability of relevance for documents binned by length for TREC 8, topics 401-450. The curve through the points represents the smoothed probabilities used for retrieval.

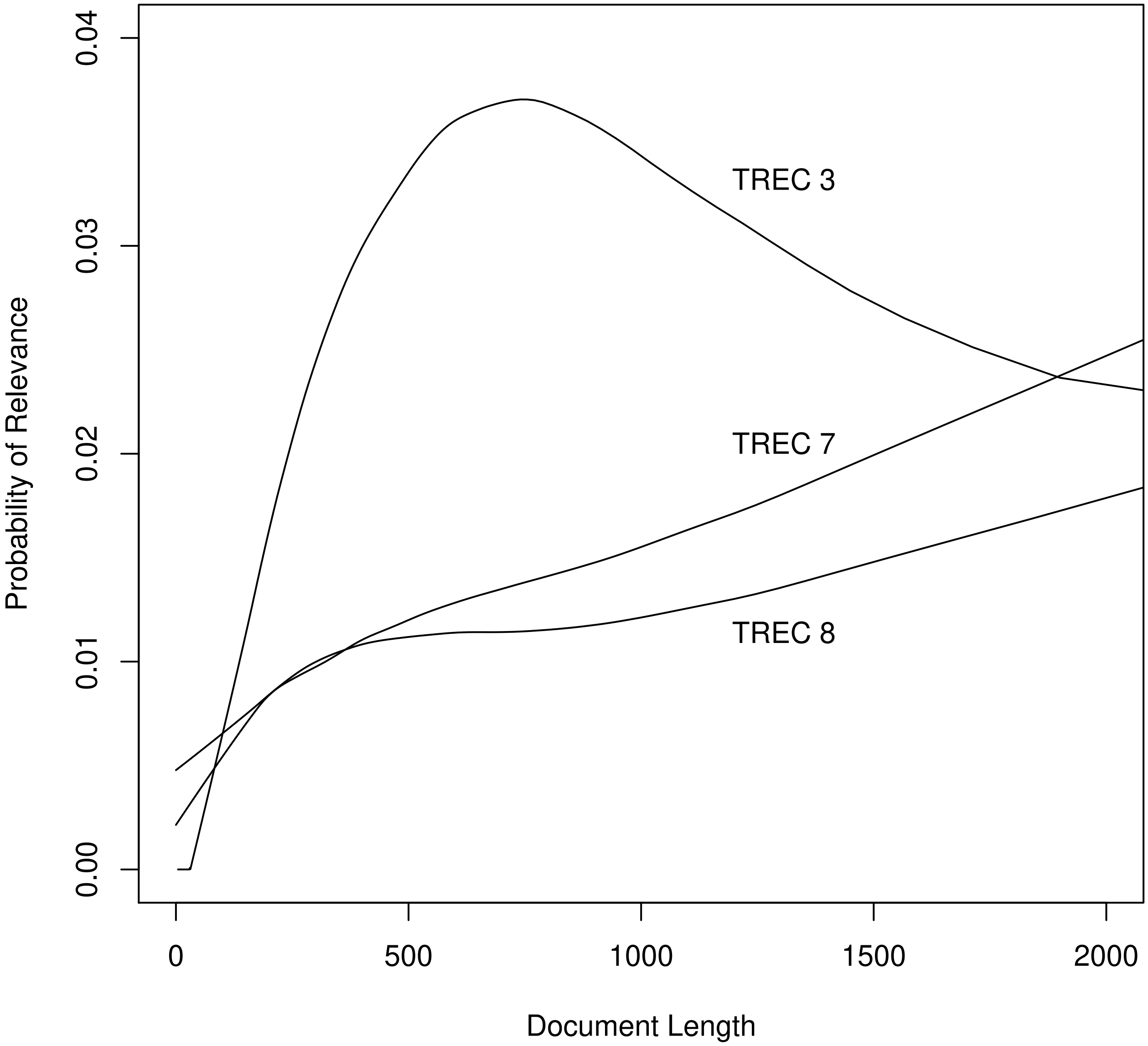

Figure 4: This figure shows a closeup of the three curves used to determine a document's prior given its length for the TREC 3, 7, and 8 datasets. TREC 7 and 8 use the same underlying collection but have different sets of relevant documents.

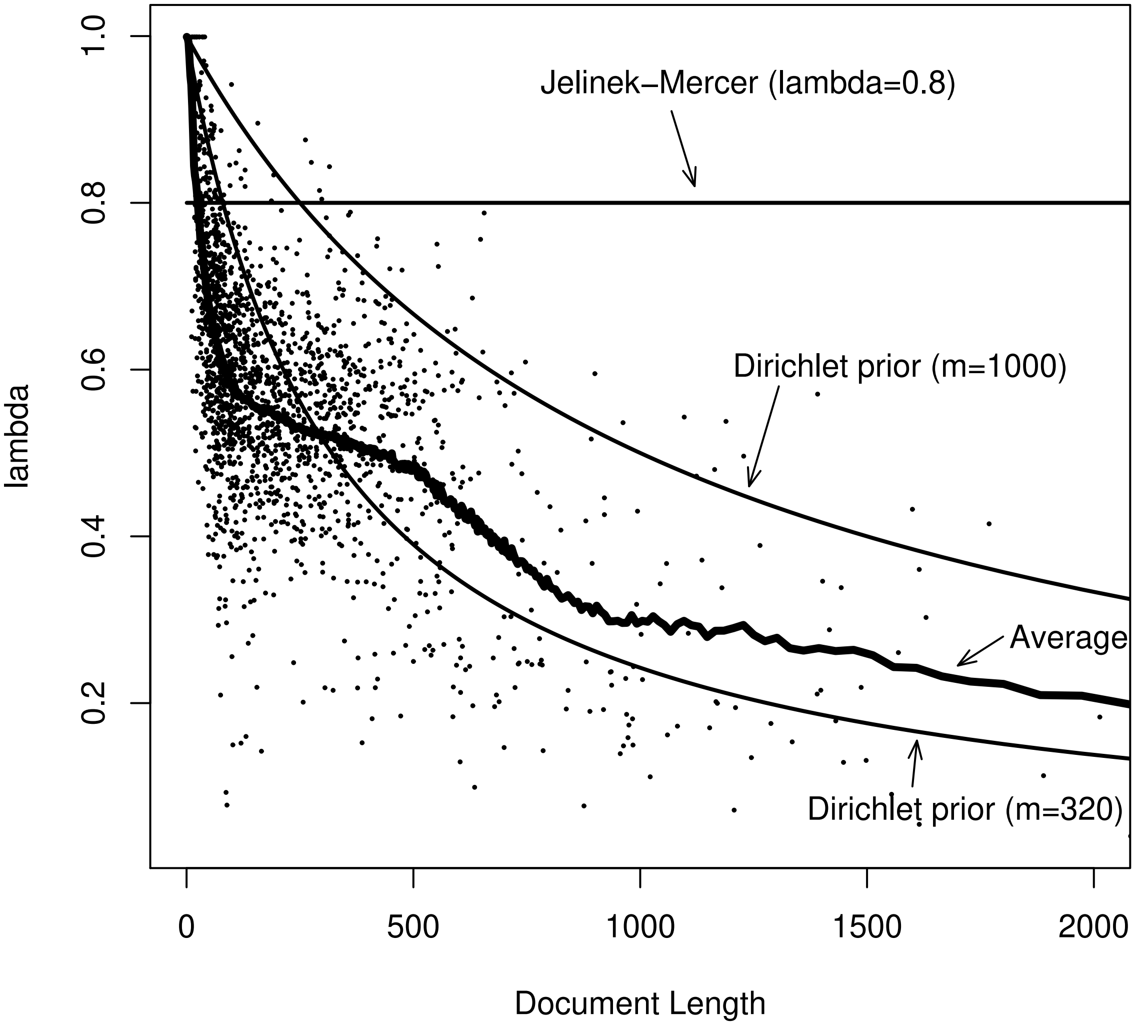

Figure 5: This figure shows the P_0 derived λ for two thousand randomly selected documents from the 1.6 million documents comprising TREC discs 1-5 minus the CR collection on discs 4 and 5. The curve marked "Average" is the average of the 1.6 million documents' P_0 derived λ's after binning the documents by length. Also shown are the equivalent λ values for the Dirichlet prior smoothing method with m set to 1000 and 320. Jelinek-Mercer smoothing is plotted with $\lambda = 0.8$.

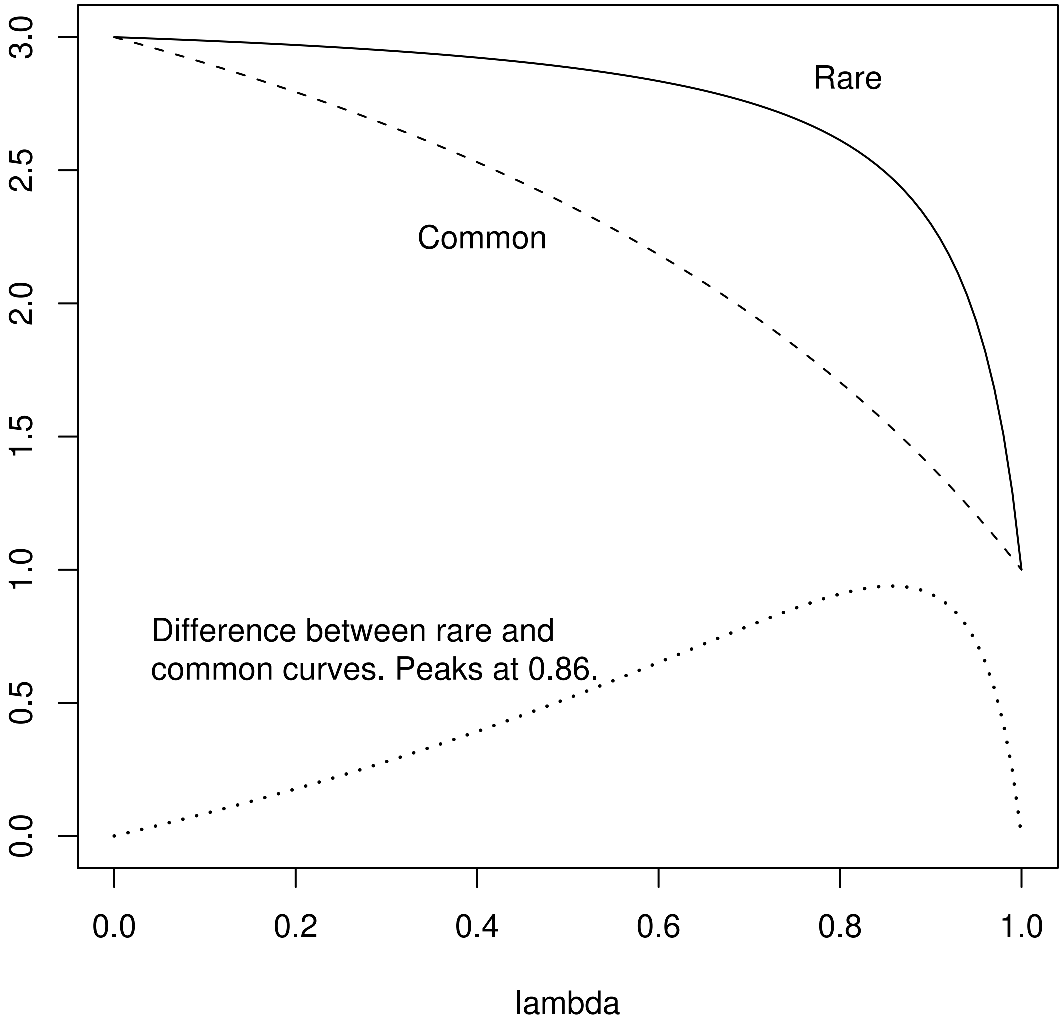

Figure 6: This figure shows an example illustrating that large amounts of linear interpolated smoothing increase the IDF effect of smoothing documents with the collection. As the linear interpolated smoothing parameter λ is increased from 0 to 1, the relative impact of the common term decreases at a faster rate than the rare term. Also plotted is the difference between the two curves. In this example, at $\lambda = 0.86$ the importance of the rare term compared to the common term is maximized.

List of Tables

			Without Prior		With Prior	
Collection	Topics	Query	JM	DP	JM	DP
Vol 1&2	151-200	title	0.217	**0.256**	**0.252**	**0.252**
Vol 4&5-CR	351-400	title	0.169	**0.190**	**0.185**	**0.187**
Vol 4&5-CR	401-450	title	0.237	**0.253**	**0.247**	**0.254**
Vol 1&2	151-200	desc.	0.183	**0.213**	**0.226**	0.204
Vol 4&5-CR	351-400	desc.	0.174	**0.189**	**0.191**	**0.177**
Vol 4&5-CR	401-450	desc.	0.224	**0.226**	**0.237**	0.217

Table 1: This table shows the non-interpolated mean average precision (MAP) scores for Jelinek-Mercer (JM) and Dirichlet prior (DP) smoothing. Scores are shown with and without a document prior that is the known probability of relevance given a document's length. JM with a prior betters or equals the performance of DP in all cases. For a given row, MAP scores in bold are statistically equivalent by a paired, two sided, Student's t-test ($p < 0.05$) compared to the highest MAP score in the row.

			JM λ		DP m	
Collection	Topics	Query	Without Prior	With Prior	Without Prior	With Prior
Vol 1&2	151-200	title	0.30	0.65	800	400
Vol 4&5-CR	351-400	title	0.55	0.80	1500	500
Vol 4&5-CR	401-450	title	0.25	0.45	350	200
Vol 1&2	151-200	desc.	0.80	0.95	1750	1500
Vol 4&5-CR	351-400	desc.	0.90	0.90	3000	1750
Vol 4&5-CR	401-450	desc.	0.85	0.90	2500	1250

Table 2: This table shows the parameter settings for both Jelinek-Mercer (JM) smoothing and Dirichlet prior (DP) smoothing that maximized the mean average precision. JM's λ increases or stays the same given the document prior while DP's m decreases in all cases.

		Given Prior from Topics			
Topics	Query	151-200	351-400	401-450	No Prior
151-200	title	**0.252**	0.234	0.226	0.217
351-400	title	0.184	**0.185**	0.180	0.169
401-450	title	0.248	0.253	**0.247**	0.237
151-200	desc.	**0.226**	0.203	0.196	0.183
351-400	desc.	0.193	**0.191**	0.186	0.174
401-450	desc.	0.239	0.242	**0.237**	0.224

Table 3: This table shows the non-interpolated mean average precision (MAP) for each set of topics and queries given different document priors. The smoothing method used is Jelinek-Mercer smoothing. The scores in bold are the same as the Jelinek-Mercer scores with a prior from Table 1.