An Investigation of Dirichlet Prior Smoothing's Performance Advantage

Mark D. Smucker, James Allan
Center for Intelligent Information Retrieval
Department of Computer Science
University of Massachusetts
Amherst, MA 01003
{smucker,allan}@cs.umass.edu

ABSTRACT

In the language modeling approach to information retrieval, Dirichlet prior smoothing frequently outperforms fixed linear interpolated (aka Jelinek-Mercer) smoothing. The only difference between Dirichlet prior and fixed linear interpolated smoothing is that Dirichlet prior determines the amount of smoothing based on a document's length. Our hypothesis was that Dirichlet prior smoothing has 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, fixed linear interpolated 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.

1. INTRODUCTION

The language modeling approach to information retrieval represents documents as generative probabilistic models [15, 13, 7, 2, 19]. A model contains a probability for each possible word. 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 construct 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 solve the zero probability problem, 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 investigated the use of three types of smoothing in information retrieval [20]. They reported on fixed linear interpolated (aka Jelinek-Mercer), Dirichlet prior (aka m-estimate), 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. Fixed linear interpolated 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 fixed linear interpolated across the nine collections. On the long queries, Dirichlet prior (DIR) was the best on six collections and fixed linear interpolated (FLI) smoothing was best on the other three, but their average performance was essentially equivalent. On the long queries, the average MAP for DIR was 0.279 vs. 0.280 for FLI. In a later work, Zhai and Lafferty reported on FLI vs. DIR performance when the sentence-length description field was used as a query [21]. Out of six collections, DIR performed better than FLI on five with an average MAP of 0.211 compared to FLI'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 fixed length interpolated smoothing.

In this paper, we attempt to answer why the Dirichlet prior (DIR) performs better than fixed linear interpolated (FLI) smoothing. As we explain later, both DIR and FLI smooth identically except that DIR determines the amount of smoothing based on a document's length. Our hypothesis is that DIR has an implicit prior that prefers longer docu-

CIIR Technical Report IR-391, January 2005.

ments, 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'll show, when FLI 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 $D : V \to \mathbb{N}$ 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 $P(w|M_D)$.

The maximum likelihood probability of a word w given a multiset X is $P(w|X)$ and is:

$$P(w|X) = \frac{X(w)}{|X|}$$

2.2 Retrieval Model

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

$$P(D|Q) = \frac{P(Q|D)P(D)}{P(Q)} \tag{1}$$

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 $P(D)$. We smooth the document D and create a document model M_D. The probability that a document generates a query, which is known as query likelihood, is given by:

$$P(Q|M_D) = \prod_{w \in Q} P(w|M_D)^{Q(w)} \tag{2}$$

Plugging Equation 2 into Equation 1 (with $P(Q)$ dropped) gives us our scoring function for a document:

$$P(D|Q) = P(D) \prod_{w \in Q} P(w|M_D)^{Q(w)} \tag{3}$$

2.3 Smoothing Methods

2.3.1 Linear Interpolated Smoothing

Linear interpolated smoothing gives the probability of a word w given a document model as follows:

$$P(w|M_D) = (1 - \lambda)P(w|D) + \lambda P(w|C) \tag{4}$$

The λ parameter is varied between 0 and 1 to control the amount of smoothing. Linear interpolated smoothing can also be written as:

$$P(w|M_D) = \lambda P(w|D) + (1 - \lambda)P(w|C) \tag{5}$$

Equation 4 is preferable and will be used in this paper since the degree of smoothing increases as λ increases from zero to one, which matches the behavior of an increase in Dirichlet prior's parameter m.

In [3] and [20], linear interpolated smoothing is termed Jelinek-Mercer. Both Manning and Schütze [12] and Jurafsky and Martin [10] refer to this as simple linear interpolation or deleted interpolation. Jelinek describes this form of smoothing as linear smoothing and uses the terminology of deleted interpolation to refer to the process of holding out data to train the parameters of the linear interpolation [8]. To avoid confusion and better express what this smoothing method is, we refer to it as linear interpolated smoothing.

2.3.2 Fixed Linear Interpolated Smoothing

As presented, linear interpolated smoothing does not vary λ based on the document being smoothed. We term this use of linear interpolated smoothing as *fixed* linear interpolated (FLI) smoothing. As we'll show next, Dirichlet prior is linear interpolated smoothing where λ is parameterized by a document's length.

2.3.3 Dirichlet Prior Smoothing

Dirichlet prior is a Bayesian justified smoothing method (see page 179 of [14]). The m-estimate is another name used to refer to Dirichlet prior smoothing. The m-estimate of probability is parameterized with a prior probability p for each word and a parameter m, known as the equivalent sample size. A document model constructed using Dirichlet prior has the probability for a word w given by:

$$P(w|M_D) = \frac{D(w) + mp}{|D| + m} \tag{6}$$

For all work in this paper, we let p be the collection model:

$$P(w|M_D) = \frac{D(w) + mP(w|C)}{|D| + m} \tag{7}$$

When estimating a document model, Dirichlet prior smoothing can be seen as starting with $|V|$ counters, one for each possible word in the collection, and initializing each counter to a value of mp and then observing the document and incrementing the counters for the words seen in the document (page 272 [1]) or as drawing m samples distributed according to p [14].

When m equals the vocabulary size and $p = 1/|V|$, Dirichlet prior becomes the smoothing method known as Add-One, which adds one to the count of each word. Church and Gale have presented an extensive argument that Add-One smoothing works poorly for language data [5]. Add-One is the same as Laplace's law of succession and is referred to as Laplace smoothing in [3] and [20]. If $m = |V|/2$ and $p = 1/|V|$ then Dirichlet prior becomes Expected Likelihood Estimation, which is also known as the Jeffreys-Perks law or

add one half. When the amount added to each count is some other value less than one, this is referred to as Lidstone's law of succession.

Dirichlet prior is a document dependent extension of linear interpolated smoothing. For each document length and m, an equivalent λ exists. Johnson showed that Lidstone's law may be viewed as linear interpolation between a maximum likelihood estimator and a uniform prior [9]. Johnson's result can be extended to include a non-uniform prior such as $P(w|C)$. This can be seen by setting λ in Equation 4 as follows:

$$\lambda = 1 - \frac{|D|}{|D| + m} \tag{8}$$

As such, one can transform the smoothing of a given document with Dirichlet prior into some equivalent linear interpolated smoothing. Because Dirichlet prior is merely linear interpolated smoothing with a λ parameterized on document length, both methods smooth a document exactly the same way for a given λ.

For example, if m is set for each document to be the document length $|D|$, the equivalent $\lambda = 1/2$, and Dirichlet prior smooths each document in an identical manner to linear interpolated smoothing with $\lambda = 0.5$. In normal practice m is fixed for a collection of documents and does not vary with document size. The result of fixing m is that λ varies and Dirichlet prior smoothing gives more weight to the prior the shorter the document is relative to m. In other words, if a document's length is less than m, it is smoothed with an equivalent λ that is greater than 0.5. If a document is longer than m, the equivalent λ is less than 0.5. Longer documents' estimates are trusted more than shorter documents' estimates.

The only difference between Dirichlet prior and FLI smoothing is that Dirichlet prior determines the amount of smoothing based on a document's length as per Equation 8. As such, the performance gains obtained by Dirichlet prior are a function of document length. Singhal et al. have reported that in the TREC collections longer documents are more likely to be relevant than short documents [18]. When a document is smoothed using linear interpolated smoothing, the probability estimates for the words in the document are moved closer to the probability of the word 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.

2.4 Document Priors

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. 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

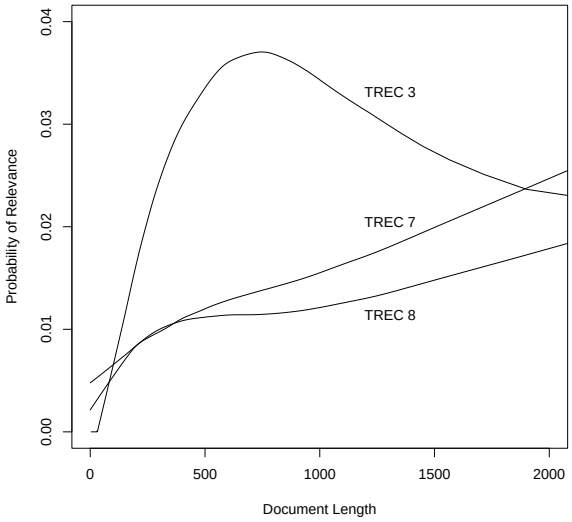

Figure 1: This figure shows a closeup of the three curves used to determine a document's prior given its length for the TREC 3, 7, and 8 datasets. TREC 7 and 8 use the same underlying collection but have different sets of relevant documents.

equal to ten [16]. 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. Figure 2 shows the computed probabilities for each bin and the smoothed curves. Our use of the document prior is inspired by Singhal et al.'s pivoted length normalization [18].

Figure 1 shows how TREC 3 has a very strong bias for relevant documents to be between 500 and 1000 tokens in length. Documents shorter than 250 tokens 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.

2.5 Experiments

The experiments consisted of determining the performance of fixed linear interpolated (FLI) and Dirichlet prior (DIR) smoothing with and without a document prior based on document length. Neither FLI nor DIR are supposed to incorporate any preference for documents based on length. A smoothing method's sole job is to provide a better estimation of a document's true probabilistic model. We know this document prior to be correct for we've calculated it directly from the known relevant documents. As such, both FLI and DIR should show performance improvements given this document prior.

If we were developing 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

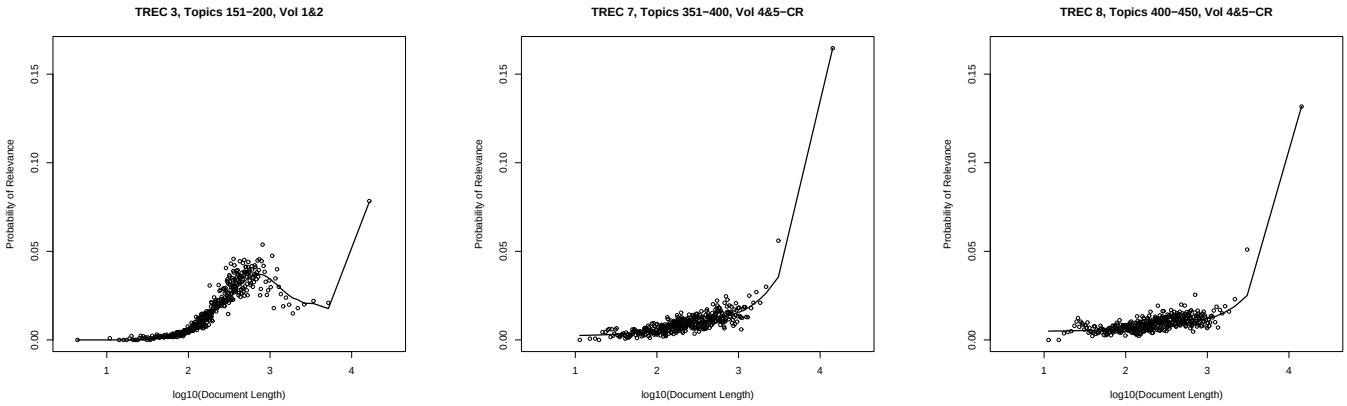

Figure 2: This figure shows plots from left to right of the computed probability of relevance for documents binned by length for TREC 3, 7, and 8. The lines through the points represents the smoothed probabilities actually used for retrieval.

are able 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. In [20], the short queries are the titles and the long queries are the concatenation of title, description and narrative fields. We agree with the formulation in [21] to use titles as keyword like non-verbose queries and descriptions as verbose queries. 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 [11] and removed stopwords using an inhouse stop word list of 418 noise words. We used Lemur 2.0.3 [22] 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 fixed linear interpolated (FLI) 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 FLI, λ was tried with values of {0.05, 0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5, 0.55, 0.6, 0.65, 0.7, 0.75, 0.8, 0.85, 0.9, 0.95}.

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 (DIR) smoothing significantly outperforms fixed linear interpolated (FLI) smoothing on every set of topics and query type. These results reproduce those of of Zhai and Lafferty [20, 21] that also show the performance advantage of DIR smoothing. In each case without the document prior, the difference between DIR and FLI is statistically significant at a $p < 0.05$ level by paired, two sided, Student's t-test.

FLI smoothing with the document prior matches or exceeds the performance of DIR smoothing both with and without the document prior in all cases. On the description queries, FLI with a prior has a better performance than DIR with a prior at a statistically significant level ($p < 0.04$) on topics 151-200 and 401-450 and is equivalent to DIR on topics 351-400. The performance of DIR smoothing with the document prior is equal to or worse than DIR smoothing without the prior.

4. DISCUSSION

Dirichlet prior (DIR) smoothing outperforms fixed linear interpolated (FLI) smoothing on both title and description queries. The only difference in smoothing between FLI and DIR smoothing is that DIR smooths longer documents less than shorter documents as per Equation 8. Both methods linearly interpolate the maximum likelihood model of the document with the collection model. We hypothesized that DIR smoothing's performance advantage arises from an implicit document prior that favors longer documents by smoothing them less. To test the hypothesis, we calculated a document prior given a document's length using the known relevant documents. We then determined the performance of both smoothing methods with and without the document prior.

When FLI smoothing is given the document prior, it matches or exceeds the performance of DIR smoothing with and without the prior. When DIR smoothing is given the doc-

			Mean Average Precision				Parameters			
			No Prior		With Prior		No Prior		With Prior	
Collection	Topics	Query	FLI	DIR	FLI	DIR	FLI λ	DIR m	FLI λ	DIR m
Vol 1&2	151-200	title	0.217	**0.256**	**0.252**	**0.252**	0.30	800	0.65	400
Vol 4&5-CR	351-400	title	0.169	**0.190**	**0.185**	**0.187**	0.55	1500	0.80	500
Vol 4&5-CR	401-450	title	0.237	**0.253**	**0.247**	**0.254**	0.25	350	0.45	200
Vol 1&2	151-200	desc.	0.183	**0.213**	**0.226**	0.204	0.80	1750	0.95	1500
Vol 4&5-CR	351-400	desc.	0.174	**0.189**	**0.191**	**0.177**	0.90	3000	0.90	1750
Vol 4&5-CR	401-450	desc.	0.224	**0.226**	**0.237**	0.217	0.85	2500	0.90	1250

Table 1: This table shows the non-interpolated mean average precision (MAP) scores and parameter settings for fixed linear interpolated (FLI) and Dirichlet prior (DIR) smoothing. Scores are shown with and without a document prior based on the probability of relevance given a document's length. FLI with a prior betters or equals the performance of DIR 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.

ument prior, its performances stays the same or worsens. In effect, DIR's implicit document prior interferes with the given document prior based on length. To compensate for the given document prior, DIR uses less smoothing than when used without the document prior. As FLI smoothing shows though, a large amount of smoothing can be used for better performance with the given document prior.

It appears that DIR's performance advantage over FLI smoothing comes more from an implicit prior against shorter documents than from better estimation of the document model.

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 FLI nor DIR 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:

$$P_0 = \frac{\mathrm{N}_1(D)}{|D|} \tag{9}$$

$\mathrm{N}_1(D)$ is the number of words that occur exactly once in the document D [17]. We will not use or discuss Good-Turing smoothing beyond using its estimation of the zero probability mass. For a good explanation of Good-Turing smoothing, we recommend [6], which is reprinted in [17].

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:

$$\sum_{w \in D} ((1-\lambda)P(w|D) + \lambda P(w|C)) = 1 - P_0$$

$$\lambda = \frac{P_0}{1 - \sum_{w \in D} P(w|C)} \tag{10}$$

A similar derivation can be done for the Dirichlet prior parameter m, but this merely produces an identical smoothing method. Using Equation 10, we can determine the amount to smooth each separate document based on the Good-Turing estimate of its zero probability mass.

Figure 3 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; shorter documents receive more smoothing than longer documents. Fixed linear interpolated (FLI) smoothing, on the other hand, smooths long and short documents equally and would be seen as a horizontal line in the figure. 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.

Zhai and Lafferty created a "leave one out" method to estimate the Dirichlet prior parameter m [21]. They find 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 distribution [22], 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 3 appears to be a reasonable fit to the P_0 derived λ values. The P_0 derived λ values imply that Dirichlet prior may smooth shorter documents too much in relation to longer documents.

4.2 IDF Behavior of Smoothing

If it is correct to smooth longer documents less and 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 does FLI smoothing with a prior succeed with high levels of smoothing for all documents on description queries? Zhai and Lafferty have shown that

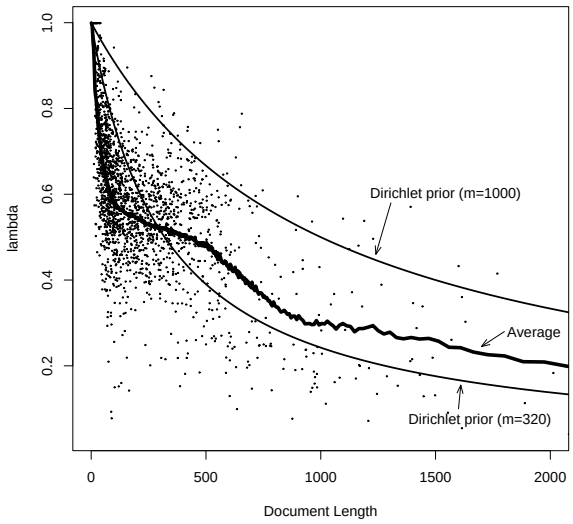

Figure 3: This figure shows the P_0 derived λ for two thousand randomly selected documents from the 1.6 million documents comprising TREC discs 1-5 minus the CR collection on discs 4 and 5. The curve marked "Average" is the average of the 1.6 million documents' P_0 derived λ's after binning the documents by length. Also shown are the equivalent λ values for the Dirichlet prior smoothing method with m set to 1000 and 320. If fixed linear interpolated smoothing were plotted, there would be a horizontal line across all document lengths at its λ setting.

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 [20]. Their and our experimental results show that longer, verbose queries require more document smoothing than short queries with DIR and FLI smoothing. As we'll illustrate with an example, high levels of smoothing increase the importance of rare terms relative to common terms. In other words, the IDF 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 2, 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:

$$\frac{P(w_1|M_A)P(w_2|M_A)}{P(w_1|M_B)P(w_2|M_B)} \tag{11}$$

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:

$$\frac{(1-\lambda)P(w_1|A) + \lambda P(w_1|C)}{(1-\lambda)P(w_1|B) + \lambda P(w_1|C)} \tag{12}$$

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 3:1 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's 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's say that the first quartile is a good representative of a rare term and the third quartile represents a common word. We thus let $P(w_1|C) = 6.0 \times 10^{-5}$ and $P(w_2|C) = 4.6 \times 10^{-4}$.

Figure 4 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 [4]. 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

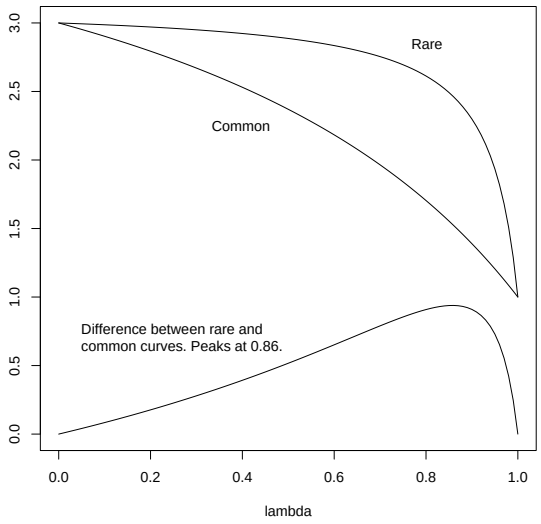

Figure 4: This figure shows an example illustrating that large amounts of linear interpolated smoothing increase the IDF effect of smoothing documents with the collection. As as the linear interpolated smoothing parameter λ is increased from 0 to 1, the relative impact of the common term decreases at 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.

ranking much faster than rare words.

The power of rare words will tend to be amplified with high levels of smoothing. This is the likely explanation of why FLI smoothing succeeds with so much smoothing even when for estimation purposes it 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.

4.3 Sensitivity to Document Priors

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 and 2 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 don't vary much based on topics but that priors for one collection may be quite different from another.

To examine the sensitivity, we used the document priors from each set of topics with the other set of topics. We only look at fixed linear interpolated (FLI) smoothing since Dirichlet prior smoothing did not show a performance improvement with the document priors. Table 2 shows the 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 2.5.

		MAP Given Prior from Topics		
Topics	Query	151-200	351-400	401-450
151-200	title	0.252	0.234	0.226
351-400	title	0.184	0.185	0.180
401-450	title	0.248	0.253	0.247
151-200	desc.	0.226	0.203	0.196
351-400	desc.	0.193	0.191	0.186
401-450	desc.	0.239	0.242	0.237

Table 2: 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 fixed linear interpolated smoothing.

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 (see Table 1).

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.

4.4 Future Work

Smoothing exists today as a monolithic component of the retrieval model performing many functions. We would like to see these functions handled by separate components of the retrieval model. Smoothing's sole role should be to produce better document estimates, but when smoothing is the only free parameter in the system, smoothing is optimized to trade off all aspects of retrieval. In this paper we've uncovered an additional role that smoothing can play as a document prior that prefers longer documents. Zhai and Lafferty previously discovered smoothing's IDF like behavior, which we've further discussed. If smoothing is playing such a big role in weighting various terms, what is the best way for IDF to be modeled in the language modeling approach? If IDF could be separately modeled like we modeled a document's prior probability given its length, then smoothing could focus on the role of estimation. We leave as future work how best to separate smoothing's roles into individual components of the retrieval model.

5. CONCLUSION

In this paper we discovered that Dirichlet prior's performance advantage over fixed linear interpolated smoothing comes more from an implicit document prior that prefers longer documents than from an ability to better estimate the true document model. We did this by constructing a prior for a given document length from the the known relevant documents. We then tested the performance of Dirichlet prior and fixed linear interpolated smoothing with and without the document prior. Both methods smooth documents

identically except Dirichlet prior smooths longer documents less. With the prior, fixed linear interpolated 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, but fixed linear interpolated smoothing's better performance on description queries seems to occur because at higher levels of smoothing, rarer terms have more influence than common terms.

6. ACKNOWLEDGMENTS

This work was supported in part by the Center for Intelligent Information Retrieval and in part by SPAWARSYSCEN-SD grant number N66001-02-1-8903. Any opinions, findings and conclusions or recommendations expressed in this material are the author(s) and do not necessarily reflect those of the sponsor.

7. REFERENCES

[1] R. K. Belew. *Finding Out About: a cognitive perspective on search engine technology*. Cambridge University Press, 2000.

[2] A. Berger and J. Lafferty. Information retrieval as statistical translation. In *Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 222–229. ACM Press, 1999.

[3] S. F. Chen and J. Goodman. An empirical study of smoothing techniques for language modeling. Technical Report TR-10-98, Center for Research in Computing Technology, Harvard University, August 1998.

[4] K. W. Church. 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*, pages 180–186. Association for Computational Linguistics, 2000.

[5] W. A. Gale and K. W. Church. What's wrong with adding one? In N. Oostdijk and P. de Haan, editors, *Corpus-Based Research into Language: In honour of Jan Aarts*, pages 189–200. Rodopi, Amsterdam, 1994.

[6] W. A. Gale and G. Sampson. Good-turing frequency estimation without tears. *Journal of Quantitative Linguistics*, 2(3):217–237, 1995.

[7] D. Hiemstra and W. Kraaij. Twenty-One at TREC-7: Ad-hoc and cross-language track. In E. M. Voorhees and D. K. Harman, editors, *The Seventh Text REtrieval Conference (TREC-7)*. Department of Commerce, National Institute of Standards and Technology, 1998. `http://trec.nist.gov/pubs/trec7/t7_proceedings.html`.

[8] F. Jelinek. *Statistical methods for speech recognition*. MIT Press, 1997.

[9] W. E. Johnson. Probability: deductive and inductive problems. *Mind*, 41(164):409–423, 1932.

[10] D. Jurafsky and J. Martin. *Speech and Language Processing*. Prentice-Hall, 2000.

[11] R. Krovetz. Viewing morphology as an inference process. In *Proceedings of the 16th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 191–202. ACM Press, 1993.

[12] C. Manning and H. Schütze. *Foundations of Statistical Natural Language Processing*. MIT Press, 1999.

[13] D. R. H. Miller, T. Leek, and R. M. Schwartz. BBN at TREC7: Using hidden markov models for information retrieval. In E. M. Voorhees and D. K. Harman, editors, *The Seventh Text REtrieval Conference (TREC-7)*. Department of Commerce, National Institute of Standards and Technology, 1998. `http://trec.nist.gov/pubs/trec7/t7_proceedings.html`.

[14] T. M. Mitchell. *Machine Learning*. McGraw-Hill, 1997.

[15] J. M. Ponte and W. B. Croft. A language modeling approach to information retrieval. In *SIGIR 1998*, 1998.

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

[17] G. Sampson. *Empirical Linguistics*, chapter 7, pages 94–121. Continuum, 2001.

[18] A. Singhal, C. Buckley, and M. Mitra. Pivoted document length normalization. In *SIGIR '96: Proceedings of the 19th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 21–29. ACM Press, 1996.

[19] F. Song and W. B. Croft. A general language model for information retrieval. In S. Gauch, editor, *Proceedings of the Eighth International Conference on Information and Knowledge Management (CIKM)*, pages 316–321. ACM, 1999.

[20] C. Zhai and J. Lafferty. A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 334–342. ACM Press, 2001.

[21] C. Zhai and J. Lafferty. Two-stage language models for information retrieval. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 49–56. ACM Press, 2002.

[22] C. Zhai, T. N. Truong, J. Lafferty, J. Callan, D. Fisher, F. Feng, J. Allan, B. Croft, P. Ogilvie, B. Jerome, A. Berger, I. Kondor, and V. Lavrenko. The lemur toolkit for language modeling and information retrieval. `http://www.cs.cmu.edu/~lemur/`.