FAIRLY RETRIEVING DOCUMENTS OF ALL LENGTHS
A STUDY OF DOCUMENT LENGTH NORMALIZATION USING THE LANGUAGE MODELING APPROACH

Leif Azzopardi
Department of Computing Science, University of Glasgow, Scotland
leif@dcs.gla.ac.uk

David E. Losada
Departamento de Electrónica y Computación, Universidad de Santiago de Compostela, Spain
dlosada@usc.es

Keywords: Information Retrieval, Document Length Normalization, Parameter Tuning, Probabilistic Language Models

Abstract: Normalizing document length is widely recognized as an important factor for adjusting retrieval systems. Previous studies have shown that tuning the retrieval model so that the lengths of retrieved documents are similar to the lengths of relevant documents will result in substantially better performance. However, the goal of Document Length Normalization is to "fairly" retrieve documents of all lengths. In this paper, we consider this proposition against the previous findings in the context of the Language Modeling approach for ad hoc information retrieval, and study the impact of the smoothing method and parameter setting on the length of documents retrieved. Our study reveals that fairly retrieving documents results in a mediocre performing parameter estimates, while using the relevant documents delivers excellent estimates. While this re-confirms previous findings, we discover that this discrepancy appears to stem from the fact that relevant documents are drawn from a biased sample, the set of assessed documents.

1 Introduction

The problem of document length normalization is ensuring that documents of particular lengths are not unduly favored over documents of other lengths by the retrieval model. The need to account for this problem is because : (1) Long documents tend to have more occurrences of different terms which means that long documents are more likely to match query terms, and; (2) As the length of the document increases the number of times a particular term occurs in the document also increases, which increases the matching score (Singhal et al., 1996). Consequently, the term weights in a document need to be penalized in accordance to document length (and thus normalize the document). By accounting for document length effects within a retrieval algorithm tends to improve performance (Singhal et al., 1996; Amati, 2003; Chowdhury et al., 2002). Although these normalization issues have been extensively studied in the context of many IR models, such as the Vector-Space model (Singhal et al., 1996), the classic Probabilistic model (Robertson et al., 1995) and Divergence From Randomness models (Amati, 2003), the effect of document length have scarcely been discussed in the Language Modeling (LM) approach.

In Language Modeling (Ponte and Croft, 1998), smoothing methods are applied to move probability mass from document terms to unseen words when constructing a LM for a document. This provides an implicit length normalization component. The smoothing level applied determines the distribution of the lengths in the retrieved set of documents, where the smoothing method and parameter dictate whether longer or shorter documents are favored. Consequently, parameter estimation is an important process as a poor estimate will bias retrieval toward long or short documents, and this may result in a serious degradation in retrieval performance. The problem is selecting the estimate that normalizes the length of documents "appropriately".

In order to determine how much normalization should be applied we re-consider the goal of document length normalization expressed by (Singhal et al., 1996): "to fairly retrieve documents of all lengths". We interpret this statement in the context of their original study, where the distribution of lengths of the retrieved document set defines a "retrieval pat-

tern". This retrieval pattern should be unbiased towards documents of particular lengths and should not unduly favor documents of one length or another, but retrieve them independently of length (only according to relevance). And so, on average, we would expect to see in the retrieved set lengths of documents that reflect the length of documents that naturally occurs in the collection. Since, we do not usually know whether the documents the user wants tend to be long, short or average in length then in general, a sensible approach is to actually remove the influence length has on the ranking of the documents, so that the retrieved set reflects the diversity of the collection.

However, in (Singhal et al., 1996), they showed that in practise retrieval patterns that were more like relevant patterns[1]. resulted in improved performance. Here, the parameters of the pivoted length normalization method for the Vector Space Model were estimated based on matching the retrieval pattern to the relevant pattern. Consequently, instead of trying to fairly retrieve documents of all lengths, the goal of normalization has been to tune the model to the relevant pattern. Because such methods have a reliance on relevance information, subsequent applications of the pivoted length normalization method have tended to resort to the default parameters suggested in the initial study (Singhal et al., 1996). However, the blanket application of these default values can result in massive loses in mean average precision, by as much as 36% in some cases (Chowdhury et al., 2002). Thus the estimated parameters may not generalize well to other queries, and more likely, not to generalize to other collections.

Consequently, an aim of the paper is to provide a retrieval model that is not biased towards documents of particular lengths, but the retrieval of relevant documents irrespective of length. We examine this idea within the LM framework, because the smoothing process not only provides an implicit document length normalization component, but because the relationship between the parameters of the model and their influence on length can be understood easily from the theory. In particular, we study the change in document lengths retrieved across parameter settings and analyze the subsequent deviations between retrieved, relevant and collection patterns.

The rest of the paper is organized as follows. The next section provides the background on LMs for retrieval and theoretically discusses how document length is affected by the two smoothing methods, Bayes Smoothing with Dirichlet Prior and Jelinek-Mercer Smoothing, and their parameter settings. Section 3 discusses the problem of document length normalization and fairly retrieving documents of all lengths, before presenting a general method for comparing the length distributions of different patterns along with a parameter estimation process. An empirical analysis is then conducted in Section 4, which examines the lengths of retrieved documents on three test collections using different length queries for each LM. Our findings are then presented and discussed in Section 5, before concluding the paper in Section 6 with a summary.

[1]The relevant pattern is defined by the distribution of document lengths defined by the set of relevant documents.

2 Background

While much attention has been paid to document length normalization with respect to the well established models, little work has examined this issue within the LM approach specifically. The standard LM approach involves scoring documents using the query likelihood with unigram document language models (Hiemstra, 2000; Zhai and Lafferty, 2001, 2004). Document scoring is essentially reduced to estimate a unigram LM for each document, $p(.|d)$, and, next, compute the probability of generating the query from each document model:

$$p(q|d) = \prod_{i=1}^{n} p(q_i|d) \tag{1}$$

Given the document's text there is a fruitful stream of research in the field of statistical natural language processing dedicated to produce robust smoothing techniques able to distribute the probability mass between the terms seen in the document and the terms missing. This is very important in LM because it is likely that a given user query will mention some non-document terms and, thus, unseen query terms in the document need to be assigned a non-zero probability. Otherwise, a single non matching term would produce a query likelihood of zero. As a consequence, a background collection, usually composed of a large number of texts, is used to define a fallback model that reflects the general use of the language and, therefore, is a good tool to smooth with.

The two smoothing techniques most often employed are the Bayesian Smoothing (BS) with Dirichlet Priors and the Jelinek-Mercer (JM) methods. In BS, the language model is defined as follows. Given a document d and a background collection C, the probability of a term in the LM of the document is computed as:

$$p(w|d) = \frac{tf(w,d) + \mu p(w|C)}{n(d) + \mu} \tag{2}$$

where $tf(w,d)$ is the raw term frequency of w in d, $n(d)$ is the total count of terms in the document (i.e.

$n(d) = \sum_w tf(w,d)$), and μ is a constant for adjusting the amount of smoothing applied. Putting together Eq. 1 and Eq. 2, applying logarithms and re-arranging terms, the retrieval score can be reduced to a simple formula involving the sum of weights for terms in common between the query and the document and a document-dependent constant (Zhai and Lafferty, 2004):

$$log\ p(q|d) \approx \sum_{i:tf(q_i,d)>0} log(1+\frac{tf(q_i,d)}{\mu p(q_i|C)}) + n\ log\frac{\mu}{n(d)+\mu} \quad (3)$$

We have therefore a retrieval formula with a penalty for long documents (second addend). The effect of this correction is higher for small μ values. As μ grows, the distinction for different lengths is less extreme and the length penalization is mitigated. However, little smoothing yields also more *coordination level ranking* (first addend)[2]. That is, small μ values prevent that a document matching n query terms can get a higher sum over matching terms than a document matching $n+1$ terms (because every single match is multiplied by $1/\mu$). Since long documents profit from coordination level strategies, there will be a point when longer documents are favored in lieu of shorter documents, as the influence from the second addendum is mitigated.

On the other hand, JM smoothing involves a linear interpolation between the maximum likelihood estimator and the probability of the term under a fallback model:

$$p(w|d) = (1-\lambda)\cdot\frac{tf(w,d)}{n(d)} + \lambda\cdot p(w|C) \quad (4)$$

The amount of smoothing applied is controlled by λ, which takes values in the interval $[0,1]$. The final retrieval score as the log query likelihood is (Zhai and Lafferty, 2004):

$$log\ p(q|d) \approx \sum_{i:tf(q_i,d)>0} \log(1+\frac{(1-\lambda)}{\lambda}\cdot\frac{tf(q_i,d)}{n(d)\cdot p(q_i|C)}) + n\cdot\log\lambda \quad (5)$$

The second addend is actually independent of any document feature and, hence, it can be ignored for ranking purposes. The final retrieval function is simply governed by a sum over matching terms. Low smoothing values ($\lambda \approx 0$) lead to a coordination level-like retrieval function whereas high λ values move away from coordination level ranking and, thus, short documents are promoted.

These two popular smoothing strategies behave differently with respect to document length, but this aspect has not been examined in depth. In (Zhai and Lafferty, 2001), several smoothing methods are analyzed for *ad hoc* IR, and Bayesian smoothing with Dirichlet priors was shown to deliver excellent retrieval performance consistently. They also found that the performance of language models are sensitive to the parameter setting for short and long queries, which required different values in order to achieve optimal performance. The study suggested some heuristics for setting λ and μ, where for short queries $\lambda \approx 0.1$ and for long queries $\lambda \approx 0.7$, while $\mu \approx 2000$ for queries of either length. However, the optimal parameter values for the different collections tended to vary. While in a study by Fang et al (Fang et al., 2004) on the heuristics of Information Retrieval, several constraints were formulated regarding these length normalization issues. The result was an estimate of a lower bound on μ, the smoothing parameter of the Bayes Smoothing with Dirichlet prior, which was equal to the average document length. In (Zhai and Lafferty, 2004), Zhai and Lafferty outlined an estimation procedure for estimating μ which is based on obtaining the best statistical representation of the document model with respect to μ. The quality of a document model given μ is obtained by computing the sum of the log likelihood of each term in the document, when it is left out (i.e. leave-one-out cross validation) over all documents. The idea is to maximize the likelihood of terms occurs in the document and so obtain the best statistical representation of the document possible. The estimate of μ tends to be slightly higher than the recommended lower bound. However, as empirical evidence has shown (Zhai and Lafferty, 2001, 2004; Fang et al., 2004), higher values of μ tend to be much more effective (i.e. more normalization is required).

3 Document Length

Both smoothing methods discussed provide some form of normalization which leads to favoring documents of different lengths, depending on the smoothing parameters. The question is what is the appropriate amount of smoothing to apply? To answer this, we resort to the goal of document length normalization put forward in (Singhal et al., 1996), that it to fairly retrieve documents of all lengths. As argued above, we interpret *fairly* as a particular document length is not preferred more or less than how often we would

[2]For a nice discussion about coordination level ranking retrieval strategies we refer to (Hiemstra, 2000).

expect to see a document of this length in the collection. For example, let us imagine that we have a collection of documents where the lengths of documents are uniformly distributed. So, we are equally likely to see a short document or a long document in the collection. If our retrieval model consistently retrieved documents, where short documents were two times more likely to be seen than long documents, then the model is biased towards short documents. In a more realistic document collection, the distribution of lengths across the collection is more like a Poisson distribution, but the intuition is the same. If the retrieval model is returning longer or shorter documents than the natural distribution then the model is biased towards the retrieval of documents of those lengths. So our goal would be to minimize the difference between the length distribution of the retrieved and the collection, on average. By doing so, we will be trying to ensure that no particular length is biased towards or against.

Later in this paper, we examine the impact of imposing this constraint in terms of retrieval performance. But first we present a general approach to study the influence of parameter estimation on any given retrieval model with respect to a predefined length distribution - so that we can examine the above proposition and compare it against alternative distributions (such as the relevant pattern).

Comparing length distributions Given a set of documents D where d is a document in the set D. The length of the document (i.e. total term occurrences) is $n(d)$. The number of documents of length k that occur in the collection is the size of the set of documents where $n(d) = k$ and defined by $\eta(n(d) = k)$. The empirical probability of a document with a length of k given the set of documents is:

$$p(L = k|D) = \frac{\eta(n(d) = k))}{n(D)} \quad (6)$$

where $n(D)$ is the number of documents in D. This probability distribution characterizes the distribution of lengths in the set. While smoothing techniques or bucketing of the lengths could be employed this would introduce more parameters, instead we avoid this requirement and compare the distributions with a metric that is robust to outliers. Hence, to determine the difference between the two length distributions we use the L1 Norm (or Least Absolute Error). The L1 Norm between two distributions defined by the set of documents D_a and D_b is the sum of the absolute difference between the probability of the length given k for each set, i.e.

$$L1(D_a, D_b) = \sum_k |p(L = k|D_a) - p(L = k|D_b)| \quad (7)$$

where $0 \leq L1(D_a, D_b) \leq 2$. The smaller the L1 Norm the closer the two distributions are, with zero indicating that the distributions are identical. The larger the L1 Norm the further the two distributions are apart, with two being the furthest distance apart. Using the L1 Norm, while a relatively simple metric, is a sound means to determine the difference between distributions, and gives much more information than a simple comparison between the means of the two distributions.

Parameter estimation using length distributions Parameters of a model can be estimated by selecting the parameter values which minimize the error between length distributions. To find this value, let D_a be the set of documents which is fixed (such as the collection or relevant documents), which we wish to use to tailor of retrieval model too. And let $D_b(\phi)$ be a set of documents which is determined by some set of parameters ϕ, (i.e. the parameter settings of the retrieval model). The objective is to find the parameter settings of the retrieval model that minimize the L1 Norm between D_a and $D_b(\phi)$,

$$\phi' = arg\ min(L1(D_a, D_b(\phi))) \quad (8)$$

The parameter set ϕ' that minimizes the error between the two distributions is selected. This is a standard mathematical optimization technique which is similar to the least squares technique. This method of estimation is applicable to any retrieval model as it relies on fitting the lengths of retrieved documents (i.e. $D_b(\phi)$) to the lengths of the optimization set (i.e. D_a).

4 Empirical Analysis

Aims In this paper, we are interested in examining the change in the distribution of lengths retrieved by different LM smoothing methods, and how this relates to the length of documents in the collection, in the pool of assessed documents[3] and the relevant documents. By changing the parameter settings of the LM the lengths of retrieved documents are affected. During the course of the study we consider three hypotheses, where the parameter value that minimizes

[3] In the context of TREC collections, these assessed documents are created by the process of pooling, which is where the top n documents from each run that participated is merged (Harman, 1995). This sub-sample of documents is then examined for relevance by an assessor.

Test Col.	Sample	N	Mean	Median
	Col.	528155	481.6	335.0
Adhoc	Ass.	86831	2861.3	681.0
TREC 8	Rel.	4728	1129.8	479.0
	Col.	247211	1055.1	368.0
WT2g	Ass.	47507	5695.6	1178.0
TREC8	Rel.	2279	5811.8	1722.0
	Col.	1033461	437.3	290.0
Aquaint (Robust)	Ass.	37799	621.0	377.0
TREC 2005	Rel.	6561	583.5	419.0

Table 1: Document Length Statistics for the entire Collection, Assessed documents, and Relevant documents in each test collection.

the error between the retrieved documents and (1) the relevant documents, (2) the document collection and (3) the assessed documents, will obtain the best performance. Since it has been previously shown that tailoring the retrieval model so that it retrieved documents of lengths similar to the relevant documents will improve the models performance, we expect that the first hypothesis will hold. The second hypothesis considers the notion of retrieving documents fairly and its impact upon performance, whilst the third examines the influence of the assessed documents.

Collections For our experiments we consider three TREC test collections, the collection used in the TREC 8 ad hoc task (Voorhees and Harman, 1999), the WT2g corpus in TREC 8 Web track (Hawking et al., 1999), and the Aquaint corpus utilized in the context of the Robust 2005 track (Voorhees, 2005). For each test collection, Table 1 reports the number of documents, and the mean and median of the documents in the set of documents, where the set is defined by all the documents in the collection (Col), by all the documents in the pool of assessed documents (Ass) and by all the documents that are relevant (Rel). Note that each set of documents is a subset of the former, i.e. $D_{Rel} \subset D_{Ass} \subset D_{Col}$. For each test collection, the $L1(D_{col}, D_{rel})$ and $L1(D_{ass}, D_{rel})$ were larger than $L1(D_{col}, D_{ass})$, i.e. the relevant documents were further from the collection and assessed, but the assessed and collection were closer together.

Method Each collection was indexed using Lemur[4]. Porter stemming was applied but no stopword processing was performed. The TREC topics were processed to produce short queries from the title field of the topics and long queries from the title and description fields. For BS, μ was set to $\{1, 10, 100, 350, 500, 1000, 1500, 2000, 3000, 5000, 10000\}$; for JM, λ was set to $\{0.01, 0.1, 0.2, \ldots, 0.9, 0.99\}$.

[4] *www.lemurproject.org*

For each collection we also ran the leave-one-out estimation procedure for BS proposed by Zhai and Lafferty (Zhai and Lafferty, 2004) which obtained $\mu_e = 623$ for the TREC 8 adhoc collection, $\mu_e = 1122$ for WT2g, and $\mu_e = 704$ for Aquaint. This parameter setting was used as one baseline to which to compare each of the hypotheses.

For each model and parameter setting we recorded the set of retrieved documents, where this set was formed by aggregating all the retrieved documents in the top 1000 documents returned in response to the queries (i.e. D_{ret})[5]. The L1 Norm between each retrieved set and the Col, Ass and Ret sets was then computed and recorded, along with the retrieval performance measurements. We then performed an analysis of the estimation procedure and the behavior of the different retrieval models. For testing statistical significance, we used the paired t-test where the significance level was set to 5% (Sanderson and Zobel, 2005).

5 Results

Figure 1 shows three subplots for JM smoothing method with short queries on the WT2g WEB TREC 8 collection with the key to all the figures. Figures 2 and 3 show all the plots for each smoothing method, and query types on the TREC 8 and Aquaint Robust 2005 collections. Similar graphs were obtained on the WT2g TREC 8 test collection but they are not all shown due to space constraints. The plots display many consistent trends and patterns of behavior for each model, which we shall now describe and discuss.

Document Length Observations In the top subplots, the mean and median lengths are shown across the parameter values which vary depending on the smoothing method employed.

In JM smoothing, increasing the length of queries increases significantly the length of documents retrieved (the right top plot shows always a higher pattern than the left top plot). This is presumably because longer documents match more query terms. But as the parameter λ tends to one, the document lengths retrieved become shorter (as expected), since longer documents are penalized according to $\frac{(1-\lambda)}{\lambda} \cdot \frac{1}{|d|}$.

In BS smoothing, increasing the length of queries, does not affect the length of the retrieved documents substantially, but longer queries tend to retrieve

[5] We also experimented with the top 100 and 500 documents and obtained similar results.

slightly longer documents. We argue that this is because of the length penalization imposed (second addend in equation 3) is multiplied by the size of the query (n). This means that longer queries impose higher penalties to long documents and, hence, this compensates for the a priori advantage of longer texts to match more query terms. It is very interesting to observe the length retrieval trends with varying μ values. Very low values of μ tend to retrieve many long documents. This means that the sum in equation 3, where every matching term is multiplied by $1/\mu$, has more influence than the penalization for long texts imposed by the second addend. That is, very low μ values lead to a retrieval strategy which is very close to a coordination level strategy and, even with the help of the second addend, a short document is unlikely to have a high rank. As μ grows this effect is compensated and less long documents are retrieved. Finally, with large μ values, the distinction between different lengths produced by the second addend is less extreme and long texts come again to populate the ranks.

In summary, the differences in behavior of the different smoothing methods stem from how they cater for the normalization of documents. Increasing λ serves to reduce the mean document length retrieved, while increasing μ serves to increase the mean document length retrieved. A general tendency observed is that the retrieval performance (in the bottom sub plot) tends to be best when the mean of retrieved documents (shown by squares) is close to the mean of the relevant documents (the dotted line). Next, we examine how close the lengths of retrieved match the three sets of documents by comparing the distance between length distributions.

Document Length Patterns On visual inspection of the L1 Norm subplots, with respect to the precision subplots, it is easy to spot that the curve defined by the $L1(ret,rel)$ reflects the precision curves (shown as squares), and so does the curve defined by the $L1(ret,ass)$. In fact, the best performance tends to be found at smoothing levels where $L1(ret,rel)$ and $L1(ret,ass)$ are minimum. This is to be expected as previous work has shown that matching the distribution of the relevant to the retrieved will yield better performance. However, for the curve defined by the $L1(ret,col)$ (shown as circles), appears to be a poorer reflection of performance. If we consider the mean length of the documents retrieved w.r.t. the collection (top plots), we notice that $L1(ret,col)$ becomes smaller as the distance between the means decreases. So whether the L1 Norm increases or decreases depends on whether the retrieved documents tend to be longer than the collection documents, in which case adjusting the model parameter so that shorter documents are retrieved will reduce the L1 Norm, and vice versa.

A very interesting pattern to witness is the intersect of the curves defined by $L1(ret,col)$ and $L1(ret,ass)$ in all L1 Norm subplots. It occurs for all plots except for JM for short queries on Trec 8 and Aquaint. The point of intersection indicates that the distance between retrieved and the collection, and the distance between the retrieved and the assessed is equal, which tends to closely correspond to where $L1(ret,rel)$ is minimized. If we examine the graphs in

Mod.	Test Col.	Method	λ/μ	mAP	Δ
JM	Adhoc T8 Short Queries	Best Empirical	**0.2**	0.2370	
		Min L1(ret,col)	0.4	0.2366	-.0004
		Min L1(ret,ass)	0.01	0.2363	-.0007
		Min L1(ret,rel)	0.01	0.2363	-.0007
	Adhoc T8 Long Queries	Best Empirical	**0.8**	0.2552	
		Min L1(ret,col)	0.9	0.2490	-.0062
		Min L1(ret,ass)	0.7	0.2502	-.0050
		Min L1(ret,rel)	**0.8**	0.2552	.0000
BS	Adhoc T8 Short Queries	Best Empirical	**500**	0.2512	
		Leave-1-out	**623**	0.2518	+.0006
		Min L1(ret,col)	350	0.2479	-.0033
		Min L1(ret,ass)	2000	0.2473	-.0039
		Min L1(ret,rel)	**500**	0.2512	.0000
	Adhoc T8 Long Queries	Best Empirical	**1500**	0.2460	
		Leave-1-out	**623**	0.2490	+.0030
		Min L1(ret,col)	500	0.2349	-.0111
		Min L1(ret,ass)	**1500**	0.2460	.0000
		Min L1(ret,rel)	1000	0.2447	-.0013

Table 2: Parameter Setting and corresponding retrieval performance obtained using each method for each model, query set on TREC 8.

Mod.	Test Col.	Method	λ/μ	mAP	Δ
JM	WT2g Short Queries	Best Empirical	**0.01**	0.2456	
		Min L1(ret,col)	0.99	0.1291*	-.1165
		Min L1(ret,ass)	**0.01**	0.2456	.0000
		Min L1(ret,rel)	**0.01**	0.2456	.0000
	WT2g Long Queries	Best Empirical	**0.1**	0.2793	
		Min L1(ret,col)	0.99	0.1434*	-.1359
		Min L1(ret,ass)	0.4	0.2670	-.0123
		Min L1(ret,rel)	0.4	0.2670	-.0123
BS	WT2g Short Queries	Best Empirical	**5000**	0.2934	
		Leave-1-out	**1122**	0.2772	-.0162
		Min L1(ret,col)	100	0.2307*	-.0627
		Min L1(ret,ass)	**5000**	0.2934	.0000
		Min L1(ret,rel)	2000	0.2858	-.0076
	WT2g Long Queries	Best Empirical	**5000**	0.3082	
		Leave-1-out	**1122**	0.2817*	-.0265
		Min L1(ret,col)	500	0.2578*	-.0504
		Min L1(ret,ass)	**5000**	0.3082	.0000
		Min L1(ret,rel)	3000	0.3021	.0061

Table 3: Parameter Setting and corresponding retrieval performance obtained using each method for each model, query set on WT2g WEB TREC 8.

Mod.	Test Col.	Method	λ/μ	mAP	Δ
JM	ROBUST Short Queries	Best Empirical	**0.5**	0.1518	
		Min L1(ret,col)	0.3	0.1509	-.0009
		Min L1(ret,ass)	0.01	0.1448	-.0070
		Min L1(ret,rel)	0.1	0.1485	-.0033
	ROBUST Long Queries	Best Empirical	**0.7**	0.1868	
		Min L1(ret,col)	0.8	0.1825	-.0043
		Min L1(ret,ass)	0.6	0.1861	-.0007
		Min L1(ret,rel)	0.5	0.1844	-.0022
BS	ROBUST Short Queries	Best Empirical	**1500**	0.2010	
		Leave-1-out	**704**	0.1947*	-.0063
		Min L1(ret,col)	100	0.1632*	-.0378
		Min L1(ret,ass)	500	0.1914*	-.0090
		Min L1(ret,rel)	500	0.1914*	-.0090
	ROBUST Long Queries	Best Empirical	**1500**	0.2149	
		Leave-1-out	**704**	0.2040*	-.0109
		Min L1(ret,col)	350	0.1881*	-.0265
		Min L1(ret,ass)	1000	0.2111	-.0071
		Min L1(ret,rel)	1000	0.2111	-0.0071

Table 4: Parameter Setting and corresponding retrieval performance obtained using each method for each model, query set on Aquaint ROBUST 2005.

Figure 3, for instance, the intersection between the assessed (triangles) and collection (circles), matches the minimum of the relevant (square), which then corresponds to the best performance. If there is no intersection, when the L1 of the assessed and collection are the closest, then it matches the minimum of the relevant. Again corresponding to the best performance.

The variation in the smoothing parameters leads to a significant variation in document length and this means that the retrieved documents move closer or further away from the collection and assessed documents, at different rates. This leads to an intersection between the two curves, of which, there are a two subcases which are divided based on the retrieval model:

IJM $L1(ret,col)$ is initially larger than $L1(ret,ass)$, but as λ increases, $L1(ret,ass)$ becomes larger than $L1(ret,col)$. That is, when we retrieve *more* long documents (smaller λ) we are closer to the assessed documents but as we retrieve *less* long documents we get closer to the collection.

IBS $L1(ret,col)$ is initially smaller than $L1(ret,ass)$, but as μ increases, $L1(ret,col)$ becomes larger than $L1(ret,ass)$. That is, when we retrieve *less* long documents (smaller μ) we are closer to the collection but when we retrieve *more* long documents we get closer to the assessed.

The assessed docs come from pools of retrieved docs and, hence, they are high ranked docs (probably relevant) and, also, long (usually, many retrieval strategies tend to favor long material). By attempting to retrieve documents fairly with respect to their length, we are assuming the length of the document has no impact on its relevance. Therefore, the distribution of relevant documents should match the distribution of documents in the collection. However, since the assessed documents is a sample of documents which tend to be much longer than the documents in the collection, the length of relevant documents is based on a biased sample. Consequently, the distribution of relevant documents will tend to be somewhere in between these two extremes. This is why there is so much of a discrepancy between the relevant and the collection, the relevant are derived from a sample which is biased - and not reflective of the collection distribution.

Usefulness for Estimation Now we examine the three hypothesis put forward. In Tables 2, 4 and 3, we report the best empirical performance for each model given the parameter values, and the performance of the model defined by the minimum L1 Norm of each set of documents, along with the performance of model when estimating μ using the leave-one-out estimation method (Zhai and Lafferty, 2004). Retrieval performance is reported by Mean Average Precision and an asterisk denotes whether the performance is significantly different from the best empirical performance.

From these Tables, we find that using the parameter value defined by the $L1(ret,rel)$ obtains near optimal performance. For the JM method this was the case on all tests, and was the case for the BS method in five out of six occasions. The exception was the BS model with short queries on the Aquaint collection. This provides evidence to confirm our first hypothesis. However for the $L1(ret,col)$ on JM the λ value tends to be larger than the best parameter setting, because it is trying to retrieve shorter documents, whereas on BS the μ values tends to be smaller than the best parameter setting, again because it is trying to retrieve shorter documents, which are more in line with the collection. Unfortunately, the mean of the relevant documents is substantially larger and thus the mismatch. On the other hand, when we consider the assessed documents, the $L1(ret,ass)$ on JM, results in λ being lower than the best parameter setting, as it is trying to retrieve longer documents. Whereas on BS, the μ value tends to larger than the parameter setting given by the minimum $L1(ret,rel)$. Using the leave-one-out estimation method to estimated μ_e, resulted performs relatively good performance. However, this method only obtained near optimal performance in three out of six occasions. In comparison, using the minimum $L1(ret,rel)$ or $L1(ret,ass)$ performed much better, with both obtaining near optimal or optimal performance five out of six times. On the other hand,

the minimum $L1(ret, col)$ resulted in the worst estimates only succeeding on two out of six occasions.

Whilst the poorer performance of the leave-one-out estimate could be attributed to the fact that the second stage of smoothing was not applied (the second stage smooths again the query to handle query variations (Zhai and Lafferty, 2002)), we also examined the performance of the models when the second stage was applied. However, the two-stage model provided more or less a uniform increase in retrieval performance regardless of the parameter setting, which was not significantly better. This results was somewhat surprising, but the L1 Norm values for the second stage were almost identical to the single stage (i.e. BS alone). This is because the second stage is only considering the query role, re-ranking documents to some extent, but not significantly affecting the length of documents retrieved. This means that the emphasis is on choosing the best estimate of μ in order to compensate for the document length problem so that optimal performance can be achieved. Since μ_e is fixed regardless of the query length, it would presumably have a negative impact. Since the length of the query affects the length of the documents retrieved (Zhai and Lafferty, 2001). Visually, we can see the difference if we inspect the document length plots using short and long queries. The longer queries retrieve shorter documents (note that this is converse for JM). Since queries of different lengths affect the length of the documents retrieved, then fixing μ or λ is not appropriate and suggests that document length normalization process is dependant on the query length. However, further work examining the influence of different sized queried is required to study the impact.

6 Conclusion

In this paper, we have considered the problem of document length normalization within the Language Modeling framework. We studied the behavior of different LMs w.r.t to document length, and reconsidered the original goal of document length normalization. By studying the lengths of documents retrieved we formed a better understanding of how the different smoothing methods operate and behave which has been insightful for parameter estimation. At this operational level, further work is possible in two main directions by (1) developing models that maintain fairness w.r.t length, but apply a document prior to fit the relevance pattern, and (2) forgoing the notion of fairness w.r.t length and building relevance patterns which the retrieval function can be specifically tailored. For instance, using the click-through data of examined documents as a surrogate relevance pattern for tuning retrieval models.

We also tested the notion that documents retrieved by the retrieval model should be distributed according to the distribution of documents in the collection. This was to ensure that the documents of any length were fairly retrieved in response to a query. While tuning the LMs in this way did not result in the best retrieval performance, we found that fitting the LMs to the relevant patterns did provide optimal or near optimal performance on most occasions. This confirms previous intuitions and findings but more importantly indicates why this is the case. When placed in context, again the collection and the assessed pattern, we can see that this may be an artifact of the test collection conditions. This is because the relevant documents are a sample from the assessed documents, and the assessed documents are significantly longer than documents in the collection. So the sample of documents which are assessed are not representative of the collection w.r.t document length. We posit that this is why adhering to the original goal of document length normalization results in sub-optimal performance. This is a key observation from this study which motivates further work to determine whether in fact there is a bias within collections or whether relevant documents are actually longer than documents in the collection.

ACKNOWLEDGEMENTS

David E. Losada thanks the support obtained from projects TIN2005-08521-C02-01 (*Ministerio de Educación y Ciencia*) and PGIDIT06PXIC206023PN (*Xunta de Galicia*). David E. Losada is funded on a "Ramón y Cajal" research fellowship, whose funds come from *Ministerio de Educación y Ciencia* and the FEDER program.

REFERENCES

Amati, G. (2003). *Divergence from Randomness*. PhD thesis, Department of Computer Science, University of Glasgow.

Chowdhury, A., McCabe, M. C., Grossman, D., and Frieder, O. (2002). Document normalization revisited. In *Proc. of the 25th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 381–382, New York, NY, USA. ACM Press.

Fang, H., Tao, T., and Zhai, C. (2004). A formal study of information retrieval heuristics. In *Proc. of the 27th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 49–56.

Harman, D. (1995). Overview of the 4th text retrieval conference. In D.Harman, editor, *Proc. TREC-4, the 4th Text Retrieval Conference*, pages 1–24. NIST.

Hawking, D., Voorhees, E., Craswell, N., and Bailey, P. (1999). Overview of the trec-8 web track. In *Proc. TREC-8, the 8th text retrieval conference*.

Hiemstra, D. (2000). A probabilistic justification for using tf x idf term weighting in information retrieval. *Int. Journal of Digital Libraries*, 3:131–139.

Ponte, J. M. and Croft, W. B. (1998). A language modeling approach to information retrieval. In *Proc. of the 21st ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 275–281.

Robertson, S., Walker, S., Jones, S., Hancock Beaulieu, M., and Gatford, M. (1995). Okapi at TREC-3. In D.Harman, editor, *Proc. of the TREC-3, the 3rd Text Retrieval Conference*, pages 109–127. NIST.

Sanderson, M. and Zobel, J. (2005). Information retrieval system evaluation: effort, sensitivity, and reliability. In *Proc. of the 28th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 162–169.

Singhal, A., Buckley, C., and Mitra, M. (1996). Pivoted document length normalization. In *Proc. of the 19th ACM SIGIR conference on Research and Development in Information Retrieval*, pages 21–29.

Voorhees, E. (2005). Overview of the trec 2005 robust retrieval track. In *Proc. TREC 2005, the 14th text retrieval conference*.

Voorhees, E. and Harman, D. (1999). Overview of the eight text retrieval conference. In *Proc. TREC-8, the 8th text retrieval conference*.

Zhai, C. and Lafferty, J. (2001). A study of smoothing methods for language models applied to ad hoc information retrieval. In *Proc. of the 24th ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 334–342.

Zhai, C. and Lafferty, J. (2002). Two-stage language models for information retrieval. In *Proc. of the 25th ACM Conference on Research and Development in Information Retrieval*, pages 49–56, Tampere, Finland.

Zhai, C. and Lafferty, J. (2004). A study of smoothing methods for language models applied to information retrieval. *ACM Trans. Inf. Syst.*, 22(2):179–214.

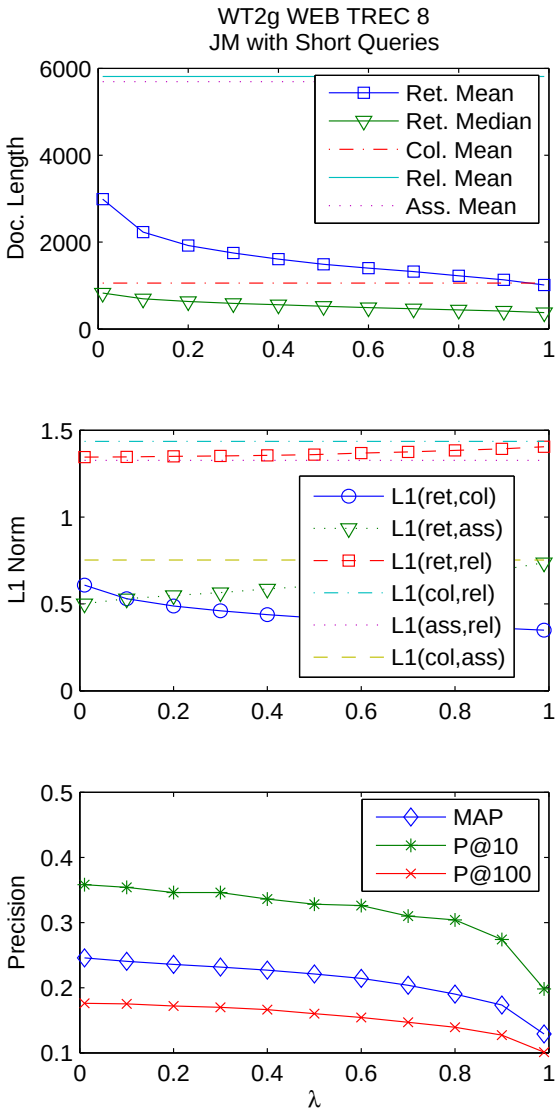

Figure 1: Example: Measures across parameter settings - WT2g WEB TREC 8 and JM with Short Queries.
The top subplot shows the change of mean (squares) and median (triangles) length of the retrieved documents over the range of parameter values. Also shown is the mean of the collection (dash dot line), assessed (dotted line) and relevant (solid line) documents.
The middle subplot, shows the change in L1 Norm between the retrieved documents and collection (circles), assessed (triangles) and relevant (squares) documents over the range of parameter values. Also shown is the L1 Norm between the collection and assessed (dashed line), the collection and the relevant (dash dot line) and the assessed and the relevant (dotted line).
The bottom subplot shows the change in mean Average Precision (mAP) (diamonds), Precision at 10 documents (P@10) (asterisks), and Precision at 100 documents (P@100) (crosses).

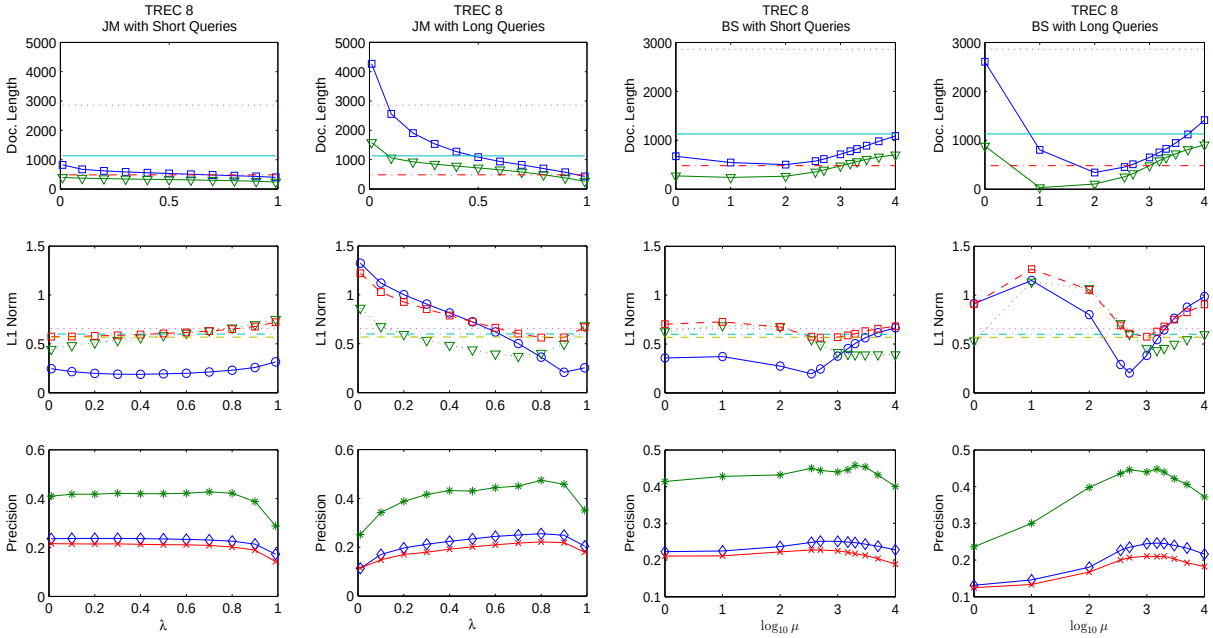

Figure 2: Measures across parameter settings - Adhoc TREC 8. Notice in the top subplots the change in mean lengths (squares) for the BS method, as either the first or second addend dominates the scoring given the parameter value. While for the JM method, the mean length decreases as $\lambda \rightarrow 1$.

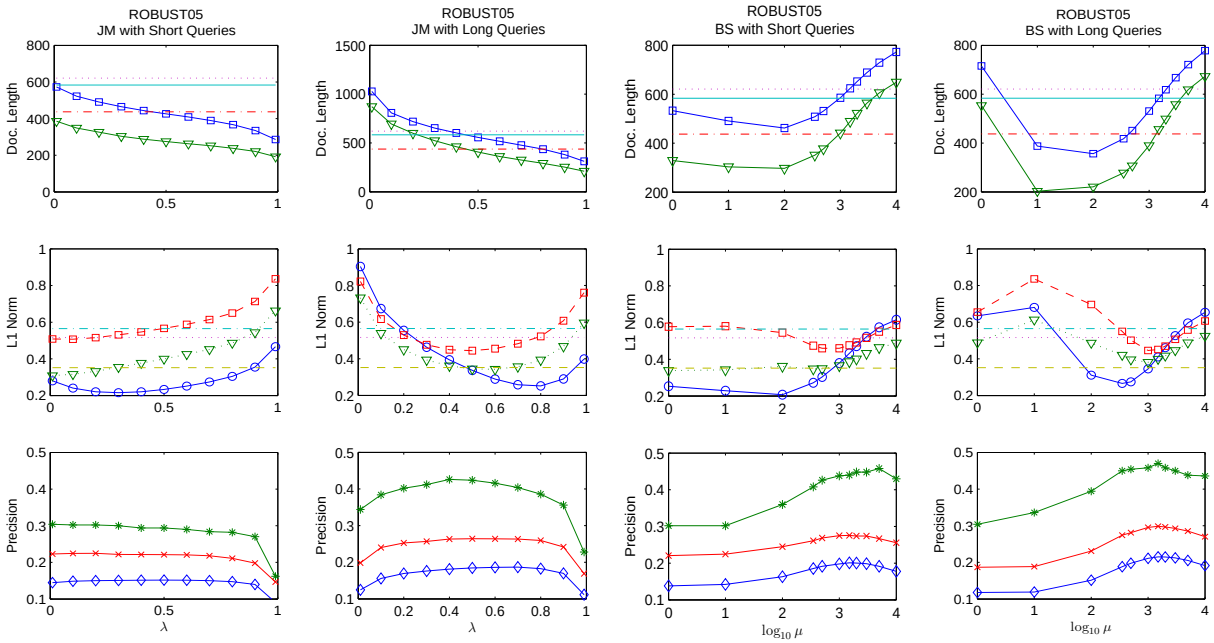

Figure 3: Measures across parameter settings - Aquaint. Notice in the middle subplots the intersection of $L1(ret, col)$ (circles) and $L1(rel, ass)$ (triangles) tend to align with the minimum $L1(ret, rel)$ (squares) . Also, how the best performance in the bottom subplots corresponds with the minimum $L1(ret, rel)$.