Probabilistic Document Length Priors for Language Models

Roi Blanco and Alvaro Barreiro

IRLab. Computer Science Department
University of A Coruña, Spain
{rblanco,barreiro}@udc.es

Abstract. This paper addresses the issue of devising a new document prior for the language modeling (LM) approach for Information Retrieval. The prior is based on term statistics, derived in a probabilistic fashion and portrays a novel way of considering document length. Furthermore, we developed a new way of combining document length priors with the query likelihood estimation based on the risk of accepting the latter as a score. This prior has been combined with a document retrieval language model that uses Jelinek-Mercer (JM), a smoothing technique which does not take into account document length. The combination of the prior boosts the retrieval performance, so that it outperforms a LM with a document length dependent smoothing component (Dirichlet prior) and other state of the art high-performing scoring function (BM25). Improvements are significant, robust across different collections and query sizes.

1 Introduction

Information retrieval (IR) systems aim to retrieve relevant documents in response to a user need, which is usually expressed as a query. The retrieved documents are returned to the user in decreasing order of relevance. Most retrieval models use term statistics, such as term frequency, to assign weights to individual terms, which represent the contribution of the term to the document content. These term weights are then used to estimate the score of relevance of a document for a query [14].

In addition to term statistics, IR models are often extended with further evidence that can improve retrieval performance, e.g. using the term frequency in specific fields of structured documents (e.g. title, abstract) [11], or integrating query-independent evidence in the retrieval model in the form of prior probabilities for a document [3,6] ('prior' because they are known before the query). In short, when determining the relevance between a query and a document, most IR models use primarily query-dependent term statistics, and sometimes also add query-independent evidence to further enhance retrieval performance. In this paper, we propose a new form of prior for documents, which we combine with IR models from the language modeling (LM) approach [8] .

C. Macdonald et al. (Eds.): ECIR 2008, LNCS 4956, pp. 394–405, 2008.

Language models for IR view documents as models and queries as segments of text generated or sampled from those models. Documents are ranked according to the probability of each query text string being generated from the respective document model. Although traditionally language models abandoned the explicit notion of document and query *relevance*, the work in [7] connects the notion of relevance and generative language models.

The LM framework models the relevance of documents to queries by estimating two probabilities (namely, query likelihood and document prior). Considering a multinomial generation of events [17], documents are ranked against queries according to those estimations. The query likelihood component is a query dependent feature representing the probability of the query being generated by the language model of a document and the document prior is a query-independent feature representing the probability of seeing the document. Typically, this probability is assumed to be the same for any document, hence the document prior is taken to be uniform [17]. Alternatively, the document prior is useful for representing and incorporating other sources of information to the retrieval process; this is currently an active area of research. For instance, document priors can be derived from the link structure of Web pages. In fact, this is a popular source for priors: [16] introduced the number of incoming links (inlinks) count, which was subsequently used in various Web retrieval tasks of the Text REtrieval Conference (TREC [15]) repeatedly and with success. Another type of evidence from which document priors are derived is URL depth, also introduced in [16]. These two priors were further explored by the work in [6]. Other URL-derived information and also the Pagerank [1] algorithm for ranking Web documents according to their popularity, have been used to derive document priors [13].

Overall, incorporating prior knowledge on documents into retrieval has been particularly effective on Web retrieval, namely *homepage* and *named page finding*. Homepage and named page finding refer to the retrieval of a single Web page; on the contrary, *ad-hoc* retrieval refers to the more general application of retrieving as much relevant information to the query as possible.

In this paper, we revisit the idea of deriving a high-quality document prior based on document length and term statistics. Most retrieval models include a document length normalisation component, so that longer documents do not have an unfair advantage over shorter documents of being retrieved. This normalisation is fairly critical and some successful models of retrieval are based in part on document length models, like BM25 [10]. We show that it is possible to encode document length information as a prior probability and improve significantly retrieval effectiveness of a simple language model that uses Jelinek-Mercer (JM) smoothing. In particular, we experiment with two length-based priors: one prior estimated proportionally to document length (we call this prior *linear* [6],[16]), and a document length based prior which is not computed directly from the number of tokens in the document but estimated in a probabilistic fashion from term statistics, which are typically used by retrieval models. To our knowledge, deriving a document prior from these term statistics is a novel approach.

Generally, priors are combined with the retrieval model either using heuristics or handtuned parameters [3]. In this work, we combine our proposed priors with the LM in two different ways: using a standard logarithmic combination, and proposing a novel combination that considers the prior as a measure of the risk of accepting the score given by the query likelihood estimation of the LM. A thorough experimentation on four TREC collections of different size and domain, and 450 short and long queries show that our proposed prior benefits retrieval performance significantly, and in a robust way. Specifically, we find that it is possible to boost the performance of a retrieval model based on JM smoothing up to values comparable to state of the art retrieval models, and further outperform retrieval models traditionally considered to be more effective in previous literature [16].

This paper is organised as follows: section 2 describes in detail the formulation used for the document priors in the rest of the paper and related work; section 3 presents a simple well-known linear document prior and a novel way of approximating document length in a probabilistic fashion; section 4 explores new ways for combining the document prior with the query likelihood, and section 5 describes our experimental findings.

2 Document Priors in the Language Modeling Approach

The language modeling framework allows a mathematically elegant way of incorporating query-independent features, i.e. just related to a document *without seeing a query*. Next, it follows a derivation of the LM retrieval model where the probability of relevance $p(r|Q, D)$, given a query and a document is estimated indirectly by invoking Bayes' rule. For the formal connection between language models and the probabilistic model of retrieval refer to [7].

Let the random variables D and Q denote a document and a query, respectively. Let the binary random variable R stand for relevance r, $p(r) = p(R = 1)$ and non-relevance $\overline{r}$, $p(\overline{r}) = p(R = 0)$.

$$\begin{aligned} p(r|Q,D) &= \frac{p(D,Q|r)\, p(r)}{p(D,Q)} && (1)\\ &= p(Q|D,r)\, p(D|r)\, \frac{p(r)}{p(D,Q)} && (2)\\ &= p(Q|D,r)\, p(r|D) \frac{p(D)}{p(D,Q)} && (3)\end{aligned}$$

Assuming independence between queries and documents $p(D, Q) = p(D)p(Q)$, and given that $p(Q)$ does not affect the ranking (it is document-independent), equation 3 becomes

$$p(r|Q,D) = \frac{p(Q|D,r)\, p(r|D)}{P(Q)} \overset{rank}{=} p(Q|D,r)\, p(r|D)\,, \quad (4)$$

where $p(Q|D, r)$ is the query likelihood and $p(r|D)$ is the document prior. In equation 4, we took a strong independence assumption to get a final formulation with dependence on $p(r|D)$. The derivation presented in [7] took a more reasonable assumption, Q and D are independent under $\overline{r}$, and starting from the odds-ratio of relevance the final relevance score is dependent on $p(r|D)/(1 - p(r|D))$.

It is usual to decompose the query into its query terms $Q = \{q_1, q_2, \ldots, q_n\}$ and assume that, given relevance and the document, they are independent of each other and generated by a multinomial distribution.

$$p(Q|D, r) = \prod_{i=1}^{n} p(q_i|D, r) \tag{5}$$

In order to rule out zero probabilities for non-seen terms in a document, this estimate has to be *smoothed*, which eventually leads to different language models-based scoring functions. Most smoothing methods employ two distributions, one for words occurring in the document (p_s) and one for *unseen* words (p_u). Taking logs (refer to [17] for a complete derivation) it can be shown that equation 6 suffices to provide a document rank using sums of logarithms, equivalent to the one that equation 5 would yield.

$$\log p(Q|D, r) \overset{rank}{=} \sum_{i \backslash tf(q_i, D) > 0} \log \frac{p_s(q_i|D)}{\alpha_d p(q_i|\mathcal{C})} + n \cdot \log \alpha_d \,, \tag{6}$$

where $tf(q_i, D)$ stands for the frequency of term q_i in document D, α_d is a parameter and $p(q_i|\mathcal{C})$ is the collection language model.

The smoothing technique we considered as our baseline in this study is Jelinek-Mercer (JM) (also known as linear interpolation):

$$p_s(q_i|D) = (1 - \lambda) p_{mle}(q_i|D) + \lambda\, p(q_i|\mathcal{C}), \;\lambda \in [0, 1]\,, \alpha_d = \lambda \tag{7}$$

where $|D| = \sum_{w_i \in D} tf(w_i, D)$ (the document length), p_{mle} is the maximum likelihood estimator for a term q_i given a document d, $p_{mle}(q_i|D) = \frac{tf(q_i,d)}{|D|}$ and λ is a parameter controlling the amount of mass distribution assigned to the *document* and *collection*.

Another popular and effective smoothing technique is Dirichlet prior smoothing:

$$p_s(q_i|D) = \frac{tf(q_i, D) + \mu p(q_i|\mathcal{C})}{|D| + \mu},\; \alpha_d = \frac{\mu}{|D| + \mu}\,, \tag{8}$$

where μ is a parameter.

In most cases $p(r|D)$ is taken to be uniform [17]. However, there have been several studies where the document length and link structure have been encoded as a prior probability, for ad-hoc and some non ad hoc tasks [6], [16].

Most weighting models include document length as a part of their core query-dependent retrieval model and that might be one of the reasons for traditionally not being considered a document static feature. For most retrieval models, the amount of normalisation contributed by document length is controlled by a parameter. This is not the case for JM smoothing, but it can be seen that the μ parameter in Dirichlet prior smoothing is playing the length normalisation role. The weight for a matched query term q_i in JM smoothing is $\log(1+(1-\lambda)p_{mle}(q_i,D)/\lambda p(q_i|\mathcal{C})))$ and for Dirichlet prior smoothing is $\log(1+|D|p_{mle}(q_i,D)/(\mu p(q_i|\mathcal{C})))$. Clearly, $|D|/\mu$ and $(1-\lambda)/\lambda$ play the same role, with the difference that the former is document-dependent while the latter is document-independent [17]. It is assumed from past studies [17],[16], that Dirichlet prior smoothing outperforms JM smoothing, especially for short queries. In our opinion, this is due to the fact that Dirichlet prior smoothing includes document length normalisation as a part of the query likelihood estimation.

Although JM smoothing does not comprise document dependent length normalisation notions, it has the advantage of "explaining" the common words of the query. This is the reason JM behaves better with long queries: these kind of queries are usually more verbose. Experiments using short-verbose queries in [17] confirmed the query-modeling role of JM smoothing. Otherwise, it is assumed that Dirichlet prior smoothing has an effect of improving the accuracy of the estimated document language model. Incorporating a good document length prior into LM-JM would hopefully result in a model that will embody both roles mentioned before.

3 Length-Based Document Priors

3.1 Linear Prior

Previous studies [12,6], tried to establish a connection between the likelihood of relevance/retrieval and document length. In particular, [12] compared the results of a set of queries and tried to obtain a relevance versus retrieval pattern (of a particular scoring function) to see how they deviate from each other. The relevance pattern happened to follow a linear dependence on document length. The results presented in [6] on a another testbed, further confirmed that hypothesis. Then, our first document length based prior is proportional to document length. The intuition behind this prior is that longer documents span more topics and are more likely to be relevant *if no query has been seen* (denoted as *scope* hypothesis in [9]). It has been reported that this prior increases the retrieval performance [6] on the WT10G collection up to 0.03 on an absolute scale.

The linear document prior is given by:

$$p(r|D) \approx \frac{|D|}{\sum_{d_i \in \mathcal{C}} |D_i|} = C \cdot |D| \,, \tag{9}$$

where C is a constant that can be dropped out from the scoring function since it does not affect the ranking of documents.

3.2 Probabilistic Prior

We propose other prior indirectly based on document length by extending the idea of estimating the document prior as a function depending on the statistics of the terms it contains.

To estimate the conditional probability $p(r|D)$ we compute the expectation over the universe of terms $\{w_i\}$. Also, in 10 we make the additional assumption that r is independent of D once we picked a term w_i.

$$p(r|D) \approx \sum_{w_i} p(r|w_i)p(w_i|D) \tag{10}$$

$$= \sum_{w_i \in D} p(r|w_i)p(w_i|D) + \sum_{w_i \notin D} p(r|w_i)p(w_i|D) \tag{11}$$

$$\approx \sum_{w_i \in D} p(r|w_i)p(w_i|D) = \sum_{w_i \in D} \left(1 - p(\overline{r}|w_i)\right) p(w_i|D) \tag{12}$$

$$= \sum_{w_i \in D} \left(1 - p(w_i|\overline{r})\frac{p(\overline{r})}{p(w_i)}\right) p(w_i|D) \approx \sum_{w_i \in D} \left(1 - p(w_i|\mathcal{C})\frac{p(\overline{r})}{p(w_i)}\right) p(w_i|D) \tag{13}$$

$$\approx \sum_{w_i \in D} p(w_i|D) \tag{14}$$

In the derivation we made the following assumptions, in order to obtain a simple model for the prior. In 11, $p(w_i|D) \approx 0$ if $w_i \notin D$. In 12 $p(w_i|\overline{r}) \approx p(w_i|\mathcal{C})$; this assumes the collection to be a model of *non-relevance*, which goes accordingly to the hypothesis taken in [4], that every document is non-relevant (and eventually leading to the inverse document frequency formula as we know it). Lastly, in 13, it is assumed for convenience that $p(\overline{r})p(w_i|\mathcal{C}) << p(w_i)$.

The final form of this prior comes from the distribution for the terms on a document, by smoothing the maximum likelihood estimator as follows:

$$p(r|D) \approx \sum_{w_i \in D} p(w_i|D) \tag{15}$$

$$= \sum_{w_i \in D} \left[(1-\lambda')\frac{tf(w_i,d)}{|D|} + \lambda' p(w_i|\mathcal{C})\right] \tag{16}$$

$$= (1-\lambda') + \lambda' \cdot \sum_{w_i \in D} p(w_i|\mathcal{C}) \tag{17}$$

In this work, it is not required that the document model employed in the prior and the document model used to compute the query likelihood be the same. The former, has a parameter, $\lambda' \in [0,1]$, coming out from the JM smoothing formula.

The result of this derivation results in a prior obtained from the sum of the individual contributions of each term occurring in the document. The linear document length-based prior (equation 9) has a similar form: it is a sum over the document terms frequencies, floored by a constant:

$$p(r|D) \approx \frac{1}{\sum_{D_i \in \mathcal{C}} |D_i|} \cdot \sum_{w_i \in D} tf(w_i, D) \tag{18}$$

The probabilistic prior is higher for documents with common terms than for documents with many rare terms, which may seem counter-intuitive. Note that the probabilistic prior counts the contribution of a term only once, despite of its document frequency. Hence, documents with many *different common* words will receive a higher prior value. Very common stopwords are likely to appear in every document, and therefore their effect is the same for every document. However, in heterogeneous collections, there may be a number of keywords describing generally its different topics or clusters. Keywords are likely to be frequent (at least inside the clusters), and documents containing many of those terms will be promoted in the rank list by the prior. This goes accordingly to the *scope* hypothesis [9]: documents covering many topics are more likely to be relevant.

4 Combination of the Prior and the Query Likelihood

In order to evaluate both priors we combine them with the query likelihood $p(Q|D,r)$ component in two different ways: a *standard* logarithmic sum and a novel method presented below. If we follow a log sum derivation from equation 4 then, the standard way of combining the document prior with the query likelihood estimation in order to produce a document score would be:

$$score(D,Q) = \log p(Q|D,r) + \log p(r|D) \tag{19}$$

We further devised a new prior-query likelihood combination, taking into account the fact that probability estimates for longer documents are more reliable than for shorter ones. We modeled this fact by considering the risk of accepting a certain score s, $\hat{R}(s) \in [0,1]$. It is possible to bias s and calculate a new score for the document and query $score(Q,D)$ as

$$score(Q,D) = s^{1-\hat{R}(s)} \tag{20}$$

Taking into account the fact that *longer* documents may provide a better estimate of $p(Q|D,r)$, it is reasonable to associate the document prior $p(r|D)$ with $1-\hat{R}(s)$, resulting in

$$score(Q,D) = scoreLM(Q,D)^{\hat{p}(r|D)} \tag{21}$$

or in logarithmic notation

$$score(Q,D) \overset{rank}{=} \hat{p}(r|D) * \log(scoreLM(Q,D)) , \tag{22}$$

where $scoreLM(Q,D)$ stands for the score a language model assigns to document D under a query Q. We combined both priors (linear and probabilistic) with the query-likelihood using this new approach. However, for the risk-based combination and linear prior, we modified the document length with a logarithmic transformation given that the probability of relevance versus logarithm of document length curve seems to be approximately linear in some ranges [12]:

$$score(Q,D) = \log(|D|) * \log(scoreLM(Q,D)) \tag{23}$$

5 Experiments and Results

The main goal of these experiments is to evaluate the effectiveness of both priors and combinations proposed before, and assess their effect on retrieval. To evaluate the new priors and combinations, we plug them into a LM with Jelinek-Mercer smoothing (equation 7). This scoring function without the prior serves as the baseline.

The TREC datasets used are described in table 1. The collections differ in size and domain, hence they represent a broad and varied experimental dataset. We experiment with short (title-only) and long (title, description and narrative) queries. We apply the standard Porter stemming algorithm, and we skip any stopwords removal, in order to avoid any bias by any choice of stoplist[1]. For all the retrieval experiments we use the Terrier IR platform[2].

The metrics used are Mean Average Precision (MAP), precision at top ten retrieved documents (P@10) and binary preference (BPref [2]). The value of the λ parameter in JM smoothing (with and without priors) has been optimised for every measure in every collection by using increasing values of 0.05 in the range (0,1]. We performed a preliminary tuning for the λ' parameter in some datasets (values increasing in 0.1 steps), and decided to set it to 0.7 for every collection. We report that it is possible to obtain marginal gains if λ' is tuned specifically for a given collection, but that step is omitted to prove the robustness of the technique.

Table 1. Collections and Topics

Collection	size	Topics	# queries
LATimes	450 MB	401-450	50
TREC disks 4&5	2G	301-450+601-700	250
WT2g	2G	401-450	50
WT10g	10G	451-550	100

The experiments presented next, compare separately the best scores produced by the two priors and two ways of combining them with the query-likelihood, with the best scores the LM-JM baseline produces. Finally, the best performing prior and combination is compared against two state of the art retrieval models (Dirichlet prior and BM25).

Table 2 presents the results for all the priors and combinations. The first column is the type of prior and combination used. JM is the baseline (without any prior). PL C1 stands for the model that uses the standard log sum combination (equation 19) and a linear prior (equation 9), and this is the only prior-combination form out of the four presented that can be found in previous studies [6]. PP C1 stands for the probabilistic prior (equation 14) and the log sum

[1] We repeated these experiments using a standard stop-word list and the conclusions derived from this experimentation are the same.

[2] http://ir.dcs.gla.ac.uk/terrier/

Table 2. Optimal performance comparison of JM with the different priors and combinations for short(left) and long(right) queries. Best values are bold. Significant MAP differences according to the Wilcoxon test ($p < 0.05$) are bold and starred.

LATimes						LATimes					
Model	MAP	P@10	Bpref	*Δ*%	p-value	Model	MAP	P@10	Bpref	*Δ*%	p-value
JM	0.2322	0.2711	0.2275	–	–	JM	**0.3010**	0.3067	**0.2865**	–	–
PL C1	0.2560	0.2889	0.2398	10.24	0.323	PL C1	0.2696	0.2978	0.2527	-10.43	0.059
PP C1	0.2332	0.2680	0.2256	0.43	0.642	PP C1	0.2937	**0.3200**	0.2848	-2.43	0.259
PL C2	0.2591	0.2784	0.2370	11.58	0.149	PL C2	0.2856	0.2978	0.2511	-5.01	0.669
PP C2	**0.2685**	**0.2889**	**0.2507**	**15.63**	**0.043***	PP C2	0.2996	0.3044	0.2861	-0.46	0.986
Disks 4&5						Disks 4&5					
Model	MAP	P@10	Bpref	*Δ*%	p-value	Model	MAP	P@10	Bpref	*Δ*%	p-value
JM	0.2333	0.3908	0.2395	–	–	JM	0.2844	0.4791	0.2838	–	–
PL C1	0.2544	0.4313	0.2583	9.04	**≈ 0***	PL C1	0.2731	0.4514	0.2741	-3.97	**0.019***
PP C1	0.2377	0.3996	0.2479	1.72	**≈ 0***	PP C1	0.2849	0.4847	0.2876	0.18	0.337
PL C2	0.2535	0.4307	0.2570	8.65	**≈ 0***	PL C2	0.2822	0.4711	0.2783	-0.77	0.537
PP C2	**0.2639**	**0.4454**	**0.2651**	**13.11**	**≈ 0***	PP C2	**0.2967**	**0.4984**	**0.2992**	**4.32**	**≈ 0***
WT2g						WT2g					
Model	MAP	P@10	Bpref	*Δ*%	p-value	Model	MAP	P@10	Bpref	*Δ*%	p-value
JM	0.2495	0.3480	0.2407	–	–	JM	0.2678	0.4300	0.2748	–	–
PL C1	0.3110	0.4660	0.2946	24.64	**≈ 0***	PL C1	0.2871	0.4660	0.2925	7.20	0.184
PP C1	0.2572	0.3760	0.2507	3.09	**0.013**	PP C1	0.2750	0.4280	0.2796	2.69	0.120
PL C2	0.3123	0.4640	0.2998	25.17	**≈ 0***	PL C2	0.3017	0.4580	0.3010	12.65	0.112
PP C2	**0.3335**	**0.4820**	**0.3182**	**33.66**	**≈ 0***	PP C2	**0.3145**	**0.4840**	**0.3138**	**17.43**	**≈ 0***
WT10g						WT10g					
Model	MAP	P@10	Bpref	*Δ*%	p-value	Model	MAP	P@10	Bpref	*Δ*%	p-value
JM	0.1479	0.2469	0.1474	–	–	JM	0.2274	0.3850	0.2202	–	–
PL C1	0.1926	0.2959	0.1889	30.22	**≈ 0***	PL C1	0.2298	0.3730	0.2338	1.05	0.592
PP C1	0.1574	0.2582	0.1597	6.42	**≈ 0***	PP C1	0.2312	0.3890	0.2291	1.67	**0.005***
PL C2	0.1939	0.3153	0.1928	31.10	**≈ 0***	PL C2	0.2366	0.3810	0.2297	4.04	0.291
PP C2	**0.1984**	**0.3316**	**0.1956**	**34.14**	**≈ 0***	PP C2	**0.2509**	**0.4020**	**0.2351**	**10.33**	**≈ 0***

combination. PL C2 stands for the linear prior and the new risk-based query likelihood combination (equation 23). Finally, PP C2 denotes the new probabilistic prior and risk-based combination (equation 22). The *Δ*% column stands for the MAP difference between the row value and the baseline. The p-value reported in the last column is obtained from the standard Wilcoxon-paired ranks sign test for the MAP results of the prior in that row and the baseline. Significant values ($p < 0.05$) are bold and starred. The best values in each column for the three measures used are bold.

Results show that under the linear combination C1, the linear prior P1 performs better for short queries whereas the probabilistic prior P2 is slightly better with long queries (in three out of four collections). Overall, improvements respect to the baseline are significant with short queries and not significant with long queries under combination C1. The risk-based combination C2 is able to improve the performance of both priors in almost every case. The behaviour of the priors changed in this case, and P2 performed better than P1 with queries of any size. In any case, the probabilistic prior under this combination always yielded the best performance among all combinations and methods tested, with some impressive improvements. Effectiveness gains are higher with shorter queries, which may be due to the fact that JM smoothing performs better for longer queries, and reduces the importance of the length normalisation step in those cases.

Table 3. Optimal performance comparison between JM+probabilistic prior, Dirichlet prior smoothing and BM25 on different collections for short(left) and long(right) queries. Best values are bold. Significant MAP differences according to the Wilcoxon test ($p < 0.05$) are bold and starred.

LATimes						LATimes					
Model	MAP	P@10	Bpref	Δ%	p-value	Model	MAP	P@10	Bpref	Δ%	p-value
JM-Prior	**0.2685**	0.2889	**0.2507**	-	-	JM-Prior	0.2996	0.3044	0.2861	-	-
BM25	0.2586	**0.2978**	0.2398	3.82	**0.041***	BM25	0.3022	0.3044	0.2870	-0.86	0.450
Dirichlet	0.2572	0.2889	0.2355	4.39	**0.017***	Dirichlet	**0.3061**	**0.3111**	**0.2970**	-2.12	0.604
Disks 4&5						Disks 4&5					
Model	MAP	P@10	Bpref	Δ%	p-value	Model	MAP	P@10	Bpref	Δ%	p-value
JM-Prior	**0.2639**	**0.4454**	**0.2651**	-	-	JM-Prior	**0.2967**	**0.4984**	**0.2992**	-	-
BM25	0.2548	0.4402	0.2565	3.57	**0.047***	BM25	0.2825	0.4896	0.2814	5.03	**0.004***
Dirichlet	0.2559	0.4329	0.2569	3.12	$\approx$ **0***	Dirichlet	0.2743	0.4667	0.2737	8.27	$\approx$ **0***
WT2g						WT2g					
Model	MAP	P@10	Bpref	Δ%	p-value	Model	MAP	P@10	Bpref	Δ%	p-value
JM-Prior	**0.3335**	**0.4820**	**0.3182**	-	-	JM-Prior	**0.3145**	**0.4840**	**0.3138**	-	-
BM25	0.3205	0.3560	0.3039	4.05	0.250	BM25	0.2833	0.4600	0.2910	11.01	**0.060***
Dirichlet	0.3087	0.4500	0.2924	8.03	**0.002***	Dirichlet	0.2906	0.4280	0.2805	8.22	**0.012***
WT10g						WT10g					
Model	MAP	P@10	Bpref	Δ%	p-value	Model	MAP	P@10	Bpref	Δ%	p-value
JM-Prior	**0.1984**	**0.3316**	**0.1956**	-	-	JM-Prior	**0.2509**	**0.4020**	**0.2351**	-	-
BM25	0.1954	0.3102	0.1872	1.53	0.45	BM25	0.2319	0.3940	0.2295	8.19	**0.012***
Dirichlet	0.1932	0.2898	0.1887	2.69	**0.035***	Dirichlet	0.2435	0.3910	0.2223	3.03	0.2708

One possible explanation for the different behaviour of both prior combinations may be due to the contribution of the prior with respect to the contribution of the query likelihood. The linear combination C1 sums the logarithm of query likelihood and prior; as the query likelihood increases (by adding more query terms) the prior contribution (query independent) diminishes. The probabilistic prior contribution does not affect much the final results when combined this particular way. A high query likelihood score is not so dominant with the risk-based combination C2: the prior is still important for the final score because the combination multiplies the prior by the query likelihood logarithm. Another result is that the effect of the prior is not very sensitive to query length with the C2 combination.

A second batch of experiments compared the new prior and combination developed in this work, probabilistic prior with the risk combination, with LM and Dirichlet prior smoothing and also against BM25. The comparison is fair, as this two matching functions already incorporate a document-dependent normalisation factor. Dirichlet prior smoothing is presented in equation 8. The μ parameter chosen is the one that optimised the performance for each metric in every collection, picked up from a reasonable set of possible choices[3]. The second weighting function considered was the probabilistic Okapi's Best Match25 (BM25) [10] which has proved to be robust, high-performing and stable in many IR studies. The behaviour of the BM25 scores is governed by three parameters, namely k_1, k_3, and b. Some studies ([5]) have shown that both k_1 and k_3 have little impact on retrieval performance, so for the rest of the paper they are set as constant to the values recommended in [10] ($k_1 = 1.2$, $k_3 = 1000$). The b parameter

[3] $\mu \in \{100, 500, 800, 1000, 2000, 3000, 4000, 5000, 8000, 10000\}$.

controls the document length normalisation factor and it has been optimised in the same way as λ for JM (parameter exploration in the $(0, 1]$ range with 0.05 steps), independently for each metric and collection. The p-values and $\Delta\%$ differences reported in table 3 are calculated considering the Dirichlet prior/BM25 run as a baseline and compared to the JM+prior (PP C2) values.

This second set of results is presented in table 3. These results prove that the PP C2 combination is able to outperform significantly high-performing retrieval matching functions in most cases (again, LATimes and long queries being the exception). We can conclude that by including a high-quality length prior, JM smoothing outperforms Dirichlet prior smoothing, which was considered superior, and also well-tuned BM25.

6 Conclusions

We developed a new document prior that takes into account term statistics and give a probabilistic derivation for it. The effect of the priors in retrieval is also dependent on the way they are combined with the query likelihood. Hence, we also demonstrated the effectiveness of a new way of combining document-length based priors with the query likelihood, that leverages the effect of the prior and likelihood components. The prior boosts the performance of a LM based on JM smoothing significantly, with robust and stable results across collections of different nature and topics of different sizes. The retrieval effectiveness of JM with the new prior is also able to outperform LM using Dirichlet prior smoothing and BM25, when the optimal parameters are used for all of them, and on the basis of three different effectiveness measures. The excellent outcome in terms of retrieval effectiveness of the prior and risk-based combination opens ground for future research directions, for instance we will try to address the problem of using this new developed way of considering document length into other retrieval matching functions, and other retrieval tasks.

Acknowledgements. We thank the reviewers for their helpful comments on the paper. This work is co-funded by FEDER, SEUI and 'Xunta de Galicia' under projects TIN2005-08521-C02 and 07SIN005206PR.

References

1. Brin, S., Page, L.: The anatomy of a large-scale hypertextual web search engine. In: WWW7: Proceedings of the seventh international conference on World Wide Web 7, pp. 107–117 (1998)
2. Buckley, C., Voorhees, E.: Retrieval evaluation with incomplete information. In: SIGIR 2004: Proceedings of the 27th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 25–32 (2004)
3. Craswell, N., Robertson, S., Zaragoza, H., Taylor, M.: Relevance weighting for query independent evidence. In: SIGIR 2005: Proceedings of the 28th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 416–423 (2005)

4. Harper, D.J., Croft, W.B.: Using probabilistic models of document retrieval without relevance information. Journal Of Documentation 35(4), 285–295 (1979)
5. He, B., Ounis, I.: A study of parameter tuning for term frequency normalization. In: CIKM 2003 Proceedings of the twelfth international conference on Information and knowledge management, pp. 10–16 (2003)
6. Kraaij, W., Westerveld, T., Hiemstra, D.: The importance of prior probabilities for entry page search. In: SIGIR 2002: Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 27–34 (2002)
7. Lafferty, J., Zhai, C.: Probabilistic relevance models based on document and query generation. In: Croft, W.B., Lafferty, J. (eds.) Language Modeling and Information Retrieval. Kluwer International Series on Information Retrieval (2002)
8. Ponte, J.M., Croft, W.B.: A language modeling approach to information retrieval. In: SIGIR 1998: Proceedings of the 21st annual international ACM SIGIR conference on Research and development in information retrieval, pp. 275–281 (1998)
9. Robertson, S.E., Walker, S.: Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In: SIGIR 1994: Proceedings of the 17th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 232–241 (1994)
10. Robertson, S.E., Walker, S., Jones, S., Hancock-Beaulieu, M.M., Gatford, M.: Okapi at TREC-3. In: Proceedings of the tenth Text Retrieval Conference (TREC-3) (1995)
11. Robertson, S., Zaragoza, H., Taylor, M.: Simple BM25 extension to multiple weighted fields. In: CIKM 2004: Proceedings of the thirteenth ACM international conference on Information and knowledge management, pp. 42–49 (2004)
12. Singhal, A., Buckley, C., Mitra, M.: Pivoted document length normalization. In: SIGIR 1996: Proceedings of the 19st annual international ACM SIGIR conference on Research and development in information retrieval, pp. 21–29 (1996)
13. Upstill, T., Craswell, N., Hawking, D.: Query-independent evidence in home page finding. ACM Transactions on Information Systems (TOIS) 21(3), 286–313 (2003)
14. van Rijsbergen, C.J.: Information Retrieval. Butterworths (1979)
15. Voorhees, E.M., Harman, D.K.: TREC: Experiment and Evaluation in Information Retrieval. The MIT Press, Cambridge (2005)
16. Westerveld, T., Kraaij, W., Hiemstra, D.: Retrieving web pages using content, links, urls and anchors. In: Proceedings of the tenth Text Retrieval Conference (TREC-10), pp. 663–672 (2002)
17. Zhai, C., Lafferty, J.: A study of smoothing methods for language models applied to information retrieval. ACM Transactions on Information Systems 22(2), 179–214 (2004)