The Sensitivity of Latent Dirichlet Allocation for Information Retrieval

Laurence A.F. Park and Kotagiri Ramamohanarao

Department of Computer Science and Software Engineering,
The University of Melbourne, 3010, Australia

Abstract. It has been shown that the use of topic models for Information retrieval provides an increase in precision when used in the appropriate form. Latent Dirichlet Allocation (LDA) is a generative topic model that allows us to model documents using a Dirichlet prior. Using this topic model, we are able to obtain a fitted Dirichlet parameter that provides the maximum likelihood for the document set. In this article, we examine the sensitivity of LDA with respect to the Dirichlet parameter when used for Information retrieval. We compare the topic model computation times, storage requirements and retrieval precision of fitted LDA to LDA with a uniform Dirichlet prior. The results show there there is no significant benefit of using fitted LDA over the LDA with a constant Dirichlet parameter, hence showing that LDA is insensitive with respect to the Dirichlet parameter when used for Information retrieval.

Keywords: latent Dirichlet allocation, probabilistic latent semantic analysis, query expansion, thesaurus.

1 Introduction

Topic models allow us to represent documents as collections of topics, rather than collections of words. It has been shown that the use of topic models for Information retrieval on large documents provides a significant increase in precision when used in an appropriate form [1,2].

Latent Dirichlet Allocation (LDA) [3] is a generative topic model that allows us to model documents using a Dirichlet prior. By changing the Dirichlet parameter, we are able to control the number of topics that the model assigns to each word and document. By setting a small Dirichlet parameter, a small number of topics are assigned to each word; by increasing the parameter, we increase the distribution of topics to each word.

The original LDA model [3] provided a means of fitting the Dirichlet parameter, when given a document set. This fitting processes requires that the document models be recomputed until the maximum likelihood Dirichlet parameter is found. Later LDA models [4,5] have avoided fitting the Dirichlet parameter by simply providing an estimate of the parameter and computing the document models once.

In this article, we investigate the effect of fitting the Dirichlet parameter for Information retrieval. We examine the change in storage and retrieval precision

W. Buntine et al. (Eds.): ECML PKDD 2009, Part II, LNAI 5782, pp. 176–188, 2009.

obtained by changing the Dirichlet parameter, and hence observe the sensitivity of the LDA topic models with respect to the Dirichlet parameter when used for Information retrieval. We make the following contributions:

- a method of computing LDA term-term relationships, allowing us to use LDA in an efficient and effective Information retrieval setting in Section 2.3, and
- a detailed comparison of document retrieval using both fitted and unfitted LDA in Section 4.

The article will proceed as follows: Section 2 describes the LDA topic model and presents a derivation of probabilistic term relationships using LDA. Section 3 presents the information retrieval model used for efficient and effective retrieval using topic models. Section 4 provides a comprehensive set of experiments comparing the effectiveness of the fitted and unfitted LDA query expansion models.

2 Probabilistic Topic Models

The language modelling method of information retrieval assumes that each document is generated from a statistical document model. To create the document, a set number of terms are sampled from the document model.

The simplest document model is a multinomial distribution across all terms, where each term has a non-zero probability of being sampled. Terms that are related to the document will have higher probabilities, while those that are not related will have low probabilities.

In this section, we will examine two popular document modelling methods, called probabilistic latent semantic analysis and latent Dirichlet allocation, that assume that the models are dependent on a set of underlying topics.

2.1 Probabilistic Latent Semantic Analysis

The simple document model mentioned assumes that each term is independent. Using this model will result in over-fitting the document set and lead to poor generalisation.

Probabilistic latent semantic analysis (PLSA) [6] introduces a set of hidden topics into the document model, so rather than directly estimating the term distribution for each document, we estimate the topic distribution over all documents and the probabilistic relationship of each document and term to each topic. PLSA uses the following document generation model:

1. To set the number of words in the document (document length), take a sample N, where $N \sim \text{Poisson}(\xi)$, and ξ is th average document length. We now have a document with N empty word slots.
2. For each of the N word slots w_i:
   (a) Select a topic z_i by sampling the distribution Multinomial(ϕ_n), conditioned on the document d_n where $P(z_i|d_n) = \phi_{i,n}$.

(b) Select a term t_x by sampling the distribution Multinomial(β_i) conditioned on the topic z_i, where $P(t_x|z_i) = \beta_{x,i}$.

Using this method of document construction, we obtain the probability of sampling term t_x in document d_n as:

$$P(t_x|d_n) = \sum_i P(t_x|z_i)P(z_i|d_n)$$

where $P(z_i|d_n) = \phi_{i,n}$ and $P(t_x|z_i) = \beta_{x,i}$. From this we can compute the set of term-term relationships [1]:

$$P(t_x|t_y) = \frac{\sum_i P(t_x|z_i)P(t_y|z_i)P(z_i)}{\sum_j P(t_y|z_j)P(z_j)} \tag{1}$$

and use these relationships for query term expansion.

2.2 Latent Dirichlet Allocation

A criticism of the PLSA approach to document modelling is that we need to compute the multinomial distribution ϕ_j which provides the probability of sampling a topic for a given document ($P(z_i|d)$). If a new document appears, we are unable to include it, since we do not have a ϕ_j distribution for it.

Latent Dirichlet allocation [3] avoids this problem by conditioning the topic selection on a Dirichlet distribution, rather than on the specific document. Latent Dirichlet Allocation (LDA) uses the following document model:

1. To set the number of words in the document (document length), take a sample N, where $N \sim$ Poisson(ξ). We now have a document with N empty word slots.
2. Given the set $\alpha = \{\alpha_1, \ldots, \alpha_k\}$, where $\alpha_i > 0$ for all i, take the sample $\theta = \{\theta_1, \ldots, \theta_k\}$, where $\theta \sim$ Dirichlet(α).
3. For each of the N word slots w_i:
   (a) Select a topic z_i by sampling the distribution Multinomial(θ).
   (b) Select a term t_x by sampling the distribution Multinomial(β_i) conditioned on the topic z_i, where $P(t_x|z_i) = \beta_{x,i}$.

We can see that rather than the topic being sampled based on a document dependent multinomial distribution, LDA samples the topic from a multinomial distribution (θ) generated from a Dirichlet space.

This gives us the likelihood function for a document:

$$P(d|\alpha, \beta) = \int \left(\prod_{n=1}^{N} \sum_i P(w_n = t_x|z_i)P(z_i|\theta) \right) P(\theta|\alpha)d\theta$$

Note that $P(w_n = t_x|z_i) = P(t_x|z_i)$, since each document is considered a bag of words. By multiplying the likelihood of each document, we obtain the likelihood of the collection. Therefore, given a document collection, we are able to compute the α and β that are most likely used to generate the document collection by maximising the likelihood function.

2.3 LDA Probabilistic Term Relationships

It has been shown that direct use of topic models for Information retrieval on large document sets lead to huge index storage requirements and slow query times [1]. A more effective and efficient method of Information retrieval using topic models is to use the knowledge of each topic in the form of a query expansion. Therefore, to use LDA in this form, we must first compute the probabilistic relationships of each term pair, given the set of topics.

Once we obtain the LDA α and β for the document set, we are able to compute the LDA relationship between each term using the following:

$$\begin{aligned} P(t_x|t_y,\alpha) &= \sum_i P(t_x, z_i|t_y,\alpha) \\ &= \sum_i P(t_x|z_i,t_y,\alpha)P(z_i|t_y,\alpha) \\ &= \sum_i P(t_x|z_i,\alpha)P(z_i|t_y,\alpha) \end{aligned} \tag{2}$$

where $P(t_x|z_i,t_y) = P(t_x|z_i)$, assuming that t_x and t_y are conditionally independent given z_i. and:

$$P(t_x|z_i,\alpha) = \beta_{x,i} \tag{3}$$

The posterior distribution is computed using Bayes' theorem:

$$\begin{aligned} P(z_i|t_y,\alpha) &= \frac{P(t_y|z_i,\alpha)P(z_i|\alpha)}{P(t_y|\alpha)} \\ &= \frac{P(t_y|z_i,\alpha)P(z_i|\alpha)}{\sum_j P(t_y,z_j|\alpha)} \\ &= \frac{P(t_y|z_i,\alpha)P(z_i|\alpha)}{\sum_j P(t_y|z_j,\alpha)P(z_j|\alpha)} \end{aligned} \tag{4}$$

From this equation, we can obtain $P(t_y|z_i,\alpha)$ from β, but we need to derive an equation for $P(z_j|\alpha)$. We know that z_i is a sample from the multinomial distribution with parameter θ, therefore we can show:

$$\begin{aligned} P(z_i|\alpha) &= \int P(z_i,\theta|\alpha)d\theta \\ &= \int P(z_i|\theta,\alpha)P(\theta|\alpha)d\theta \\ &= \int P(z_i|\theta)P(\theta|\alpha)d\theta \end{aligned} \tag{5}$$

where $P(z_i|\theta,\alpha) = P(z_i|\theta)$, assuming that z_i and α are conditionally independent given θ.

Since $P(z_i|\theta) = \theta_i$, we can simply replace the probability function in equation 5 with a function of $g_i(\theta)$ that simply returns the ith element of θ:

$$\begin{aligned} P(z_i|\alpha) &= \int g_i(\theta) P(\theta|\alpha) d\theta \\ &= \mathbb{E}_{P(\theta|\alpha)}[g_i(\theta)] \\ &= \mathbb{E}_{P(\theta|\alpha)}[\theta_i] \end{aligned}$$

We know that $\theta \sim \text{Dirichlet}(\alpha)$, therefore:

$$\mathbb{E}_{P(\theta|\alpha)}[\theta_i] = \frac{\alpha_i}{\sum_k \alpha_k} \tag{6}$$

By substituting equation 6 into equation 4, we obtain:

$$\begin{aligned} P(z_i|t_y, \alpha) &= \frac{P(t_y|z_i, \alpha)\frac{\alpha_i}{\sum_k \alpha_k}}{\sum_j P(t_y|z_j, \alpha)\frac{\alpha_j}{\sum_k \alpha_k}} \\ &= \frac{P(t_y|z_i, \alpha)\alpha_i}{\sum_j P(t_y|z_j, \alpha)\alpha_j} \end{aligned}$$

and finally we substitute equation 3 to obtain:

$$P(z_i|t_y, \alpha) = \frac{\beta_{y,i}\alpha_i}{\sum_j \beta_{y,j}\alpha_j} \tag{7}$$

By substituting equation 7 into 2, we can obtain a closed solution for our term relationships:

$$\begin{aligned} P(t_x|t_y, \alpha) &= \sum_i P(t_x|z_i, \alpha) P(z_i|t_y, \alpha) \\ &= \sum_i \beta_{x,i} \frac{\beta_{y,i}\alpha_i}{\sum_j \beta_{y,j}\alpha_j} \\ &= \frac{\sum_i \beta_{x,i}\beta_{y,i}\alpha_i}{\sum_j \beta_{y,j}\alpha_j} \end{aligned} \tag{8}$$

Therefore, we are able to obtain the probabilistic relationships between each term using the α and β values computed by fitting the LDA model to our document collection. It is also easy to see the equivalence of equation 8 and the PLSA term relationships in equation 1.

2.4 Exchangeable and Uniform Dirichlet Distribution

An exchangeable Dirichlet prior implies that any of the Dirichlet prior parameters can be exchanged without affecting the distribution. To achieve this, an exchangeable Dirichlet prior must contain all equal elements.

When using an exchangeable Dirichlet prior ($\alpha_i = a$ for all i), the Dirichlet distribution simplifies to:

$$\begin{aligned} P(\theta|\alpha) &= \frac{\Gamma(\sum_{i=1}^{k} \alpha_i)}{\prod_{i=1}^{k} \Gamma(\alpha_i)} \prod_{i=1}^{k} \theta_i^{\alpha_i - 1} \\ &= \frac{\Gamma(\sum_{i=1}^{k} a)}{\prod_{i=1}^{k} \Gamma(a)} \prod_{i=1}^{k} \theta_i^{a-1} \\ &= \frac{\Gamma(ka)}{\Gamma(a)^k} \prod_{i=1}^{k} \theta_i^{a-1} \\ &= \frac{\Gamma(ka)}{\Gamma(a)^k} \left(\prod_{i=1}^{k} \theta_i \right)^{a-1} \end{aligned}$$

A further specialisation of the Dirichlet prior is the Uniform Dirichlet prior ($\alpha_i = 1$ for all i), which is an exchangeable Dirichlet with $a = 1$:

$$\begin{aligned} P(\theta|\alpha) &= \frac{\Gamma(k)}{\Gamma(1)^k} \left(\prod_{i=1}^{k} \theta_i \right)^{0} \\ &= \frac{\Gamma(k)}{1^k} \\ &= \Gamma(k) \\ &= (k-1)! \end{aligned}$$

We can see that the probability is independent of the value of θ and therefore uniform across all values of θ.

To examine the term relationships produced when using an exchangeable Dirichlet distribution, we add the constraint that $\alpha_i = a$ for all i. By doing so, then we obtain the term relationship probabilities:

$$\begin{aligned} P(t_x|t_y, \alpha) &= \frac{\sum_i \beta_{x,i} \beta_{y,i} \alpha_i}{\sum_j \beta_{y,j} \alpha_j} \\ &= \frac{\sum_i \beta_{x,i} \beta_{y,i} a}{\sum_j \beta_{y,j} a} \\ &= \frac{\sum_i \beta_{x,i} \beta_{y,i}}{\sum_j \beta_{y,j}} \end{aligned} \tag{9}$$

Therefore, when using an exchangeable Dirichlet prior for LDA, the term relationship probabilities become independent of α. When examining the term relationship probabilities using a uniform Dirichlet prior, we obtain the same equation. Note that although α does not appear in equation 9, the values of β are still affected by the choice of α, therefore we would not obtain the same term relationships for when choosing either an exchangeable or uniform Dirichlet prior.

When using a uniform Dirichlet prior, the topic distribution obtains a uniform prior and can be modelled using a multinomial distribution. It has been shown that under this condition, LDA is equivalent to probabilistic latent semantic analysis (PLSA) [7].

3 Topic Based Query Expansion

Information retrieval systems obtain their fast query times by building a fast query term lookup index over a sparse set of term frequencies. When a query is provided, the index lists corresponding to the query terms are extracted and combined to obtain the document scores. If term frequencies are stored, the index becomes sparse, and hence we will be able to process the index lists efficiently.

When using topic models such as LDA, every term has a probabilistic relationship to every document, and hence the data is not sparse. If we were to store these values in the document index, we would obtain a large dense index containing long index lists that would take substantial time to process.

Rather than store the topic model probabilities in the document index, it was found that it is more efficient to store these values as term relationships in a thesaurus and store the term frequencies in the document index [1,8,9,10]. By doing so, we obtain a compact document index for fast retrieval and control over the query time by using the topic models in the form of of a thesaurus for query expansion.

By using the index-thesaurus combination, we compute each document score using:

$$s_d(Q) = \mu \mathrm{S}_d(E) + (1 - \mu)\mathrm{S}_d(Q)$$

where Q is the set of query terms, E is the set of weighted expansion terms computed by applying the query terms to the topic model thesaurus (as shown in [1,2]), $\mathrm{S}_d()$ is the document scoring function for document d and $\mu \in [0, 1]$ is the query mixing parameter to combine the original query terms to the expansion terms.

In the following experiments, we chose $\mathrm{S}_d()$ to be the state-of-the-art BM25 probabilistic document scoring function [11].

4 Comparison of Fitted and Unfitted LDA Term Expansion

Now that we have shown how to compute the LDA term relationships using the LDA topic models, we will proceed to investigate their effectiveness. In this section, we will examine the precision obtained by LDA and examine if there is a statistically significant increase when fitting the Dirichlet parameter.

To build the LDA thesaurus, we first compute the set of $\beta_{x,i}$ values for the given document set using the LDA-C software[1] (which assumes an exchangeable

[1] http://www.cs.princeton.edu/∼blei/lda-c/

Table 1. Statistics related to the LDA thesauruses used to store the probabilistic term relationships. Note that the Dirichlet parameter was set to the value of $\alpha = 1$, making the unfitted form of LDA equivalent to PLSA.

Document set	File size		Build time		Dirichlet α	
	Fitted	Unfitted	Fitted	Unfitted	Fitted	Unfitted
AP2	112 MB	87 MB	2 days	57 minutes	0.020	1
FR2	30 MB	29 MB	1 day	17 minutes	0.021	1
WSJ2	107 MB	90 MB	2.5 days	61 minutes	0.022	1
ZIFF2	52 MB	42 MB	2 days	32 minutes	0.033	1

Dirichlet prior) and then compute the thesaurus values using equation 9. Our experiments will compare the LDA with the Dirichlet parameter fitted to the document set (fitted LDA) to LDA with a uniform Dirichlet prior ($\alpha = 1$) representing the unfitted LDA.

Experiment 1: Effect of the number of topics and terms on Average Precision. The LDA thesaurus build time is dependent on the number of terms and the number of topics. Therefore our first set of experiments will examine the effect of changing the number of terms and topics on the retrieval average precision. Initial experiments compared the retrieval results using LDA term expansion when using 1) all terms and 100 topics, 2) 100 topics and terms that appear in at least 50 documents, and 3) 300 topics and terms that appear in at least 50 documents. Average precision (AP) results were obtained using the ZIFF2 document set from TREC[2] disk 2, with queries 51 to 200 (used with disk 2 for TRECs 1, 2 and 3).

We generated AP scores from the 150 queries for mix values of 0.0 to 1.0 in intervals of 0.1 and term expansions of 10, 20, 50, 100, 200, 500 and 1000. By applying the two-sided Wilcoxon Signed Rank test to our AP results, we found that there was no significant difference in the AP of any of the three forms of LDA term expansion. These results are consistent with the findings when using PLSA term expansion [1]. Based on these results, we will now only consider LDA term expansion using 100 topics and only terms that appear in at least 50 documents.

Experiment 2: Thesaurus storage and build time. Before performing any retrieval experiments, we must first construct the thesaurus containing the probabilistic term relationships.

Table 1 shows the thesaurus statistics corresponding to the fitted and unfitted forms of LDA term relationships for the AP2, FR2, WSJ2 and ZIFF2 document sets from TREC disk 2. We can see from this table that the fitted thesaurus file sizes are larger than the corresponding unfitted thesaurus, and that the term

[2] http://trec.nist.gov

Table 2. Mean average precision on FR2 using BM25 with fitted LDA term expansion. Single and double superscript daggers († and ††) show a significant increase in precision over unfitted LDA, using the same mix and expansion size, at the 0.1 and 0.05 levels respectively. Single and double subscript stars (∗ and ∗∗) show a significant increase in precision over BM25. Scores in *italics* show the greatest score for the given expansion size.

Mix	Expansion size						
	10	20	50	100	200	500	1000
0.0	0.197	0.197	0.197	0.197	0.197	0.197	0.197
0.1	0.198_{**}	0.198_{**}	0.198_{**}	0.198_{**}	0.199_{**}	0.203_{**}	0.203_{**}
0.2	0.198_{**}	0.198_{**}	0.203_{**}	0.203_{**}	0.204_{**}	0.204_{**}	0.204_{**}
0.3	0.198_{**}	0.199_{**}	0.204_{**}	0.204_{**}	0.205_{**}	0.206_{**}	0.206_{**}
0.4	0.202_{**}	0.204_{**}	0.204_{**}	0.206_{**}	0.209_{**}	0.210_{**}	0.210_{**}
0.5	0.203_{**}	0.204_{**}	0.207_{**}	0.209_{**}	0.212_{**}	0.214_{**}	0.216_{**}
0.6	0.204_{**}	0.205_{**}	0.209_{**}	0.212_{**}	0.215_{**}	0.218_{**}	0.220_{**}
0.7	0.207_{**}	0.210_{**}	0.213_{**}	0.215_{**}	0.221_{**}	0.222_{**}	0.227_{**}
0.8	$\mathit{0.209}_{**}$	$\mathit{0.212}_{**}$	$\mathit{0.215}_{**}$	$\mathit{0.223}_{**}$	$\mathit{0.226}_{**}$	$\mathit{0.228}_{**}$	$\mathit{0.233}_{**}$
0.9	0.207_{**}	0.205_{**}	0.211_{**}	0.212_{*}	0.213	0.219_{*}	0.219_{*}
1.0	0.018	0.023	0.031	0.037	0.036	0.037	0.043

Table 3. Mean average precision on ZIFF2 using BM25 with fitted LDA term expansion. Single and double superscript daggers († and ††) show a significant increase in precision over unfitted LDA, using the same mix and expansion size, at the 0.1 and 0.05 levels respectively. Single and double subscript stars (∗ and ∗∗) show a significant increase in precision over BM25. Scores in *italics* show the greatest score for the given expansion size.

Mix	Expansion size						
	10	20	50	100	200	500	1000
0.0	0.269	*0.269*	0.269	0.269	0.269	0.269	0.269
0.1	0.269_{*}	$\mathit{0.269}_{*}$	0.269_{**}	0.269_{**}	0.269_{**}	0.269_{**}	0.269_{**}
0.2	0.269_{**}	$\mathit{0.269}_{**}$	0.269_{*}	0.269	0.269	0.269_{*}	0.269
0.3	$0.269^{\dagger}$	$0.268^{\dagger}$	0.268	0.268	0.271	0.270	0.270
0.4	0.269	0.268	0.268	0.271	0.270	0.270	0.270
0.5	0.268	0.268	*0.271*	*0.272*	0.272	0.272	0.272
0.6	*0.270*	0.268	0.270	0.271	0.272	0.273	$0.273^{\dagger}$
0.7	0.269	0.269	0.270	0.270	0.273	0.272	$\mathit{0.280}^{\dagger\dagger}$
0.8	0.267	$0.268^{\dagger}$	$0.270^{\dagger}$	$0.271^{\dagger\dagger}$	$\mathit{0.278}^{\dagger}$	$\mathit{0.278}^{\dagger}$	$0.278^{\dagger}$
0.9	$0.257^{\dagger\dagger}$	$0.257^{\dagger\dagger}$	$0.259^{\dagger\dagger}$	$0.264^{\dagger\dagger}$	$0.256^{\dagger\dagger}$	$0.264^{\dagger\dagger}$	$0.253^{\dagger\dagger}$
1.0	0.014	0.016	0.015	0.015	$0.014^{\dagger}$	0.015	0.015

Table 4. Mean average precision on AP2 using BM25 with fitted LDA term expansion. Single and double superscript daggers († and ††) show a significant increase in precision over unfitted LDA, using the same mix and expansion size, at the 0.1 and 0.05 levels respectively. Single and double subscript stars (∗ and ∗∗) show a significant increase in precision over BM25. Scores in *italics* show the greatest score for the given expansion size.

Mix	Expansion size						
	10	20	50	100	200	500	1000
0.0	0.271	0.271	0.271	0.271	0.271	0.271	0.271
0.1	0.272_{**}	0.272_{**}	0.272_{**}	0.272_{**}	0.272_{**}	0.272_{**}	0.272_{**}
0.2	0.272_{**}	0.272_{**}	0.273_{**}	0.273_{**}	0.273_{**}	0.273_{**}	0.273_{**}
0.3	0.273_{**}	0.273_{**}	0.274_{**}	0.275_{**}	0.275_{**}	0.275_{**}	0.275_{**}
0.4	0.274_{**}	0.274_{**}	0.275_{**}	0.275_{**}	0.276_{**}	0.276_{**}	0.276_{**}
0.5	0.274_{**}	0.275_{**}	0.277_{**}	0.277_{**}	0.278_{**}	0.278_{**}	0.278_{**}
0.6	0.275_{**}	0.276_{**}	0.278_{**}	0.279_{**}	0.279_{**}	0.280_{**}	0.280_{**}
0.7	0.276_{**}	0.277_{**}	0.280_{**}	0.280_{**}	0.281_{**}	0.282_{**}	0.282_{**}
0.8	0.277_{**}	0.279_{**}	$\mathit{0.282}_{**}$	$\mathit{0.283}_{**}$	$\mathit{0.284}_{**}$	$\mathit{0.284}_{**}$	$\mathit{0.284}_{**}$
0.9	$\mathit{0.279}^{\dagger\dagger}_{**}$	$\mathit{0.281}^{\dagger}_{**}$	$0.281^{\dagger\dagger}_{**}$	$0.282^{\dagger\dagger}_{**}$	$0.282^{\dagger\dagger}_{**}$	$0.283^{\dagger\dagger}_{**}$	$0.282^{\dagger\dagger}_{**}$
1.0	0.030	0.031	0.037	0.039	0.039	0.038	0.037

relationships take substantially longer to compute (due to the fitting process). The table also shows the fitted α value, demonstrating that the computed fitted LDA term relationships are different from the computed unfitted LDA term relationships.

Experiment 3: Comparison of Average Precision when using fitted and unfitted LDA query expansion. Our next set of experiments compare the fitted LDA term expansion precision to unfitted LDA term expansion and the baseline BM25 (no term expansion). Again, we will use mixing values of 0.0 to 1.0 in intervals of 0.1 and term expansions of 10, 20, 50, 100, 200, 500 and 1000 for both fitted and unfitted LDA. For this set of experiments we will use the AP2, FR2, WSJ2 and ZIFF2 document sets from TREC disk 2 with queries 51-200. For each of the document sets, thesauruses were built for fitted and unfitted LDA using 100 topics and only those terms that appear in at least 50 documents. The mean average precision results are shown in Tables 2, 3, 4 and 5 for the documents sets FR2, ZIFF2, AP2 and WSJ2 respectively.

We can see from table 2 that fitted LDA did not provide a significant increase in precision over unfitted LDA for any values of mix or expansion size on FR2. Also, table 3 shows that fitted LDA did not provide a significant increase in precision over BM25 for most mix values and expansion sizes on ZIFF2.

The AP2 and WSJ2 results in tables 4 and 5 show that the fitted LDA term expansion provides the greatest mean average precision (MAP) at $\mu = 0.8$. We can also see that the MAP increases as the query expansion size is increased.

Table 5. Mean average precision on WSJ2 using BM25 with fitted LDA term expansion. Single and double superscript daggers († and ††) show a significant increase in precision over unfitted LDA, using the same mix and expansion size, at the 0.1 and 0.05 levels respectively. Single and double subscript stars (∗ and ∗∗) show a significant increase in precision over BM25. Scores in *italics* show the greatest score for the given expansion size.

Mix	Expansion size						
	10	20	50	100	200	500	1000
0.0	0.257	0.257	0.257	0.257	0.257	0.257	0.257
0.1	0.257_{**}	0.257_{**}	0.258_{**}	0.258_{**}	0.258_{**}	0.258_{**}	0.258_{**}
0.2	0.258_{**}	0.258_{**}	0.258_{**}	0.258_{**}	0.258_{**}	0.258_{**}	0.259_{**}
0.3	0.258_{**}	0.258_{**}	0.258_{**}	0.259_{**}	0.259_{**}	0.259_{**}	0.259_{**}
0.4	0.258_{**}	0.259_{**}	0.259_{**}	0.259_{**}	0.260_{**}	0.262_{**}	0.262_{**}
0.5	0.259_{**}	0.259_{**}	0.259_{**}	0.259_{**}	0.262_{**}	0.263_{**}	0.263_{**}
0.6	0.259_{**}	0.259_{**}	0.260_{**}	0.263_{**}	0.264_{**}	0.265_{**}	0.265_{**}
0.7	0.262_{**}	0.260_{**}	0.261_{**}	0.264_{**}	0.266_{**}	0.267_{**}	0.268_{**}
0.8	$\mathit{0.262}_{**}$	$\mathit{0.262}_{**}$	$\mathit{0.264}_{**}$	$\mathit{0.267}_{**}$	$\mathit{0.268}_{**}$	$\mathit{0.270}_{**}$	$\mathit{0.270}_{**}$
0.9	$0.260^{\dagger}$	$0.259^{\dagger}$	$0.260^{\dagger}_{*}$	$0.262^{\dagger\dagger}_{**}$	$0.263^{\dagger\dagger}_{**}$	$0.260^{\dagger\dagger}_{**}$	$0.260^{\dagger\dagger}_{**}$
1.0	0.025	0.025	0.029	0.031	0.033	0.032	0.032

Note that a mix of $\mu = 0$ implies that there are no expansion terms used and hence the score is simply the BM25 score.

Each of the tables contain information about statistical significance tests. For each run, we have reported statistically significant increases in Average Precision (AP) using the one sided Wilcoxon signed rank test at the 0.1 and 0.05 levels of fitted LDA over each of unfitted LDA (using the associated expansion size and mix) and BM25. We can see from both the AP2 and WSJ2 results that the fitted LDA thesaurus provides a significant increase in average precision over BM25 for most results where the mix values are between $\mu = 0.1$ and 0.9. We can also see that fitted LDA provides a significant increase over unfitted LDA for a mix of $\mu = 0.9$ only for both AP2 and WSJ2.

Given that fitted LDA provides the greatest mean average precision with a mix of $\mu = 0.8$, but does not provide a significant increase in precision over unfitted LDA for both of the AP2 and WSJ2 document set, we can deduce that we have not gained any precision advantages from fitting the Dirichlet parameter.

Table 6 contains the mix-expansion pair that provide the greatest MAP for each method on each data set. Also provided are the precision at 10 and mean reciprocal rank scores. This table shows little difference between the retrieval performance of fitted and unfitted LDA.

From this evidence we have obtained, we deduce that the extra computation required to compute the fitted LDA topic relationships do not provide any benefit over the unfitted LDA topic relationships for the all of the document sets examined.

Table 6. A comparison of the mean average precision (MAP), precision at 10 (Prec10) and mean reciprocal rank (MRR) of each system on each data set. The mix-expansion combination shown are those that produce the greatest MAP.

Data	Method	Mix	Expansion	MAP	Prec10	MRR
AP2	Fitted LDA	0.8	200	0.284	0.380	0.562
	Unfitted LDA	0.8	200	0.282	0.382	0.563
	BM25	N/A	N/A	0.271	0.355	0.537
FR2	Fitted LDA	0.8	1000	0.233	0.141	0.368
	Unfitted LDA	0.8	1000	0.234	0.141	0.367
	BM25	N/A	N/A	0.197	0.117	0.314
WSJ2	Fitted LDA	0.8	500	0.270	0.370	0.625
	Unfitted LDA	0.8	500	0.269	0.369	0.625
	BM25	N/A	N/A	0.257	0.353	0.572
ZIFF2	Fitted LDA	0.7	1000	0.280	0.169	0.427
	Unfitted LDA	0.8	500	0.280	0.165	0.460
	BM25	N/A	N/A	0.269	0.158	0.415

5 Conclusion

Latent Dirichlet allocation (LDA) is a generative topic model that allocates topics using the Dirichlet distribution. To compute the topic distribution, we need to obtain the Dirichlet parameter for the document set, which can be either fitted using maximum likelihood, or estimated.

In this article, we examined the effect of fitting the Dirichlet parameter with the LDA topic model on the precision for Information retrieval. To do so, we derived an expression for LDA term-term probabilistic relationships for use as a query expansion.

We compared the effectiveness of query expansion using fitted and unfitted LDA on several document sets and found that using fitted LDA did not provide a significant increase in Average Precision when operating under its peak settings (mix=0.8). We showed that by not fitting the Dirichlet parameter, we obtained a 50 to 90 times gain in computational efficiency when computing the topic models. Considering that the fitted LDA term relationships consume more storage and time to build, we conclude that fitting the Dirichlet parameter provides no advantage when using LDA for an Information retrieval task, hence showing that LDA is insensitive with respect to the Dirichlet parameter for Information retrieval.

References

1. Park, L.A.F., Ramamohanarao, K.: Efficient storage and retrieval of probabilistic latent semantic information for information retrieval. The International Journal on Very Large Data Bases 18(1), 141–156 (2009)

2. Park, L.A.F., Ramamohanarao, K.: An analysis of latent semantic term self- correlation. ACM Transactions on Information Systems 27(2), 1–35 (2009)
3. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent dirichlet allocation. Journal of Machine Learning Research 3, 993–1022 (2003)
4. Griffiths, T.L., Steyvers, M.: Finding scientific topics. Proceedings of the National Academy of Sciences of the United States of America 101(suppl. 1), 5228–5235 (2004)
5. Wei, X., Croft, W.B.: LDA-based document models for ad-hoc retrieval. In: SIGIR 2006: Proceedings of the 29th annual international ACM SIGIR conference on Research and development in information retrieval, pp. 178–185. ACM, New York (2006)
6. Hofmann, T.: Probabilistic latent semantic indexing. In: Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval, pp. 50–57. ACM Press, New York (1999)
7. Girolami, M., Kaban, A.: On an equivalence between PLSI and LDA. In: SIGIR 2003: Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval, pp. 433–434. ACM Press, New York (2003)
8. Park, L.A.F., Ramamohanarao, K.: Query expansion using a collection dependent probabilistic latent semantic thesaurus. In: Zhou, Z.-H., Li, H., Yang, Q. (eds.) PAKDD 2007. LNCS (LNAI), vol. 4426, pp. 224–235. Springer, Heidelberg (2007)
9. Park, L.A.F., Ramamohanarao, K.: Hybrid pre-query term expansion using latent semantic analysis. In: Rastogi, R., Morik, K., Bramer, M., Wu, X. (eds.) The Fourth IEEE International Conference on Data Mining, pp. 178–185. IEEE Computer Society, Los Alamitos (2004)
10. Park, L.A.F., Ramaohanarao, K.: The effect of weighted term frequencies on probabilistic latent semantic term relationships. In: Amir, A., Turpin, A., Moffat, A. (eds.) SPIRE 2008. LNCS, vol. 5280, pp. 63–74. Springer, Heidelberg (2008)
11. Jones, K.S., Walker, S., Robertson, S.E.: A probabilistic model of information retrieval: development and comparative experiments, part 2. Information Processing and Management 36(6), 809–840 (2000)