Bayesian Folding-In with Dirichlet Kernels for PLSI

Alexander Hinneburg
Martin-Luther-University
Halle-Wittenberg, Germany
hinneburg@informatik.uni-halle.de

Hans-Henning Gabriel
101tec GmbH
Mansfelder Str. 13
06108 Halle, Germany
hg@101tec.com

Andrè Gohr
Leibniz Institute for Plant
Biochemistry, IPB, Germany
agohr@ipb-halle.de

Abstract

Probabilistic latent semantic indexing (PLSI) represents documents of a collection as mixture proportions of latent topics, which are learned from the collection by an expectation maximization (EM) algorithm. New documents or queries need to be folded into the latent topic space by a simplified version of the EM-algorithm. During PLSI-Folding-in of a new document, the topic mixtures of the known documents are ignored. This may lead to a suboptimal model of the extended collection.

Our new approach incorporates the topic mixtures of the known documents in a Bayesian way during folding-in. That knowledge is modeled as prior distribution over the topic simplex using a kernel density estimate of Dirichlet kernels. We demonstrate the advantages of the new Bayesian folding-in using real text data.

1. Introduction and Context

The representation of documents as mixture proportions of latent topics is proven to be a useful tool for text mining. PLSI [7, 9] opened the way for probabilistic modeling of such representations. Applications and extensions of the PLSI model in the field of data mining include author-topic identification [14], textmining [11], and web usage mining [10].

A major drawback is, that PLSI is susceptible to overfitting [13]. The following research explored different Bayesian extensions of the PLSI model itself as well as alternative undirected models to overcome different drawbacks. Bayesian extensions include latent Dirichlet allocation (LDA) [4], which models the topic mixture proportions of a document as hidden variables drawn from a single Dirichlet distribution, rather than as parameters of the model. PLSI has been shown to be a special case of LDA [6], when it uses an uninformative flat Dirichlet as prior. A more sophisticated variant are Dirichlet process priors [1]. Another Bayesian extension of PLSI are correlated topic models [2], which use a log-normal prior distribution. Such a distribution can capture correlations between topics, which cannot be expressed by a single Dirichlet. Dynamic topic models [3] extent this framework to model temporal changes in the latent topic space.

Alternative models from the class of undirect graphical models are undirected PLSI [15] and the Rate Adapting Poisson (RAP) model [5]. Those models are trained using contrastive divergence and avoid drawbacks of Bayesian directed models like the explaining away effect.

A general drawback of the proposed extensions and alternatives to PLSI is, that the improvements come at the price of increased runtime costs for the inference algorithms, which hinders an applications to large data.

A more direct approach to learning of document similarities are Fisher information kernels [8], which make use of the latent decomposition of the term-document matrix and model the similarities between pairs of documents directly. A further improvement is reported in [12].

PLSI is not a generative model, so a special procedure called folding-in has to be used to get the topic mixture proportions of new documents or queries. Our approach extends PLSIs folding-in in a Bayesian way instead of extending the PLSI model itself. Instead of using a maximum likelihood estimator for folding-in, a maximum a posteriori estimator is employed, which uses a kernel density estimate as prior. The used kernel is a Dirichlet density. The advantage of a kernel density estimate as prior is, that only a very few model assumptions are made. Also such a prior can express correlations between topics similar to the correlated topic model. The contributions of the paper are: (i) we propose a new Bayesian model for folding-in, and (ii) a new inference technique is introduced which uses EM to maximize the posterior consisting of the word likelihood and the kernel density prior.

The reminder of the paper is structured as follows, in section 2 we elaborate on the problems of PLSIs folding-in. In section 3, we introduce the new Bayesian model for folding-

in. Next, we present in section 4 an applications of how to use our new model. Last, we describe our experiments on real text data in section 5 and conclude the paper.

2. Problems of PLSIs Folding-In

Let $\mathcal{D}$ be a collection of N documents $\mathcal{D} = \{d_1, \ldots, d_N\}$, and each document is represented by a bag-of-words, which is a subset of the vocabulary of size V. PLSI [7, 9] models the co-occurrence of documents and words as a mixture of K latent classes $P(d, w) = P(d) \sum_{j=1}^{K} P(w|a_j)P(a_j|d)$. The parameters of the model are the topic-word associations $\vec{\omega} = [\omega_{ij} = P(w_i|a_j)]_{i=1,\ldots,V,j=1,\ldots,K}$ and the document-topic mixtures $\vec{\theta} = [\theta_{lj} = P(a_j|d_l)]_{l=1,\ldots,N,j=1,\ldots,K}$, which are estimated by an EM-algorithm. A K-dimensional row $\vec{\theta}_l$ of the latter matrix denotes the mixture of topics for document l.

PLSI is not a generative model, thus the topic mixture $P(a|d_q)$ is not known for some new query document d_q. The proposed folding-in procedure [7, 9] estimates those topic mixtures by running the original EM with fixed word-topic associations. So, the folding-in procedure reduces to (only the underlined probabilities are allowed to change during the algorithm):

$$\text{E-step: } P(a_j|w_i, d_q) = \frac{P(w_i|a_j)\underline{P(a_j|d_q)}}{\sum_{j'=1}^{K} P(w_i|a_{j'})\underline{P(a_{j'}|d_q)}} \quad (1)$$

$$\text{M-step: } P(a_j|d_q) = \frac{\sum_{i=1}^{V} n(d_q, w_i)\underline{P(a_j|w_i, d_q)}}{n(d_q)} \quad (2)$$

The quantities $n(d_q, w_i)$ and $n(d_q)$ are the number of occurrences of word w_i in d_q and the number of words in d_q respectively.

Note, that the topic mixture for d_q is found independently from the mixtures of the other documents in the collection. The only influence comes through the fixed word-topic associations $P(w|a)$, which are involved in the right-hand side of the E-step.

This can lead to problems in case of short queries, which do not contain a rich vocabulary as the documents in the collection. Since those queries have much fewer words with non-zero frequencies as the documents and the raw frequency counts are one in most cases, PLSIs folding-in tends to produce topic mixtures which are dominated by a single latent aspect.

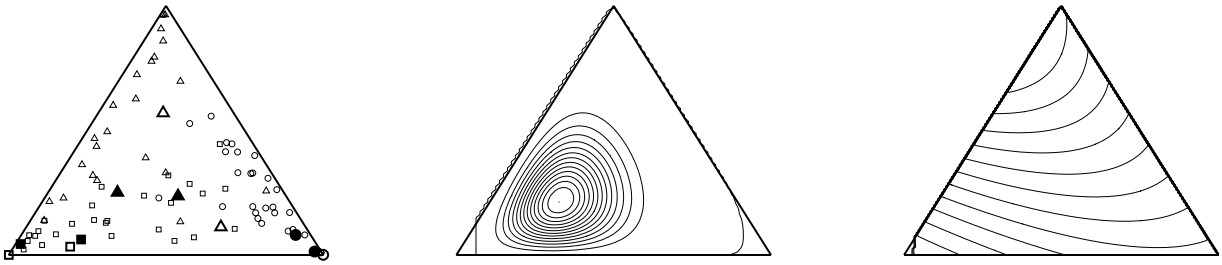
The following example demonstrates this behavior. The data consists of $28 + 28 + 28 = 84$ documents, which are randomly sampled from three different newsgroups of the *20newsgroups collection*. Stopwords as well as infrequent words are eliminated and the other words are reduced to their stemmed form by Porters stemmer. PLSI run with $K = 3$ latent aspects maps the documents of the three different newsgroups (small circles, squares and triangles) into the latent space shown by the simplex in figure 1(a). The six big filled icons represent topic mixtures of documents from the respective groups found by PLSIs folding-in. The other six big empty icons simulate short queries each consists of four words sampled from one of the folded-in documents. Figures 1(b) and (c) show the likelihoods induced by the left most filled big triangle and the upper most big empty triangle query respectively. While the long document induces a likelihood with a clear local maximum, the likelihood of the short version of that query has its maximum close to the upper corner of the simplex. Only the tempered version of EM [7] used for folding prevents that the short query is mapped to that border position. However, note the empty big circles and squares representing the other short queries in the left and right corners of the simplex in figure 1(a), where the tempered EM could not help. Such a corner position indicates that only a single latent aspect is present in such a query, which, however, in case of short queries is mainly caused by the small sample of words in the query. So, PLSIs folding-in cannot account for alternative mappings of such a query, which might correspond to alternative semantic interpretations of the query.

3. Bayesian Folding-In

We present a new Bayesian way to estimate the mixture proportions of topics $\vec{\theta}_q$ for a new (query) document d_q with the word vector $\vec{w}_q = (w_1, \ldots, w_M)$. Instead of maximizing the likelihood $P(\vec{w}_q|\vec{\theta}_q, \vec{\theta}, \vec{\omega})$ of the word vector $\vec{w}_q$ with respect to $\vec{\theta}_q$, the posterior $P(\vec{\theta}_q|\vec{w}_q, \vec{\theta}, \vec{\omega})$ is maximized. This maximum a posteriori (MAP) approach requires the definition of a prior distribution for the mixture of topics $\vec{\theta}_q$ of the new (query) document.

Because the topic mixtures of the documents in the collection shall have some influence on the topic mixture of the query, the prior is modeled as kernel density estimate using a Dirichlet distribution as kernel function. Kernel density estimation is a quite flexible method, which does not make strong assumptions about the distribution of the topic mixtures of the documents in the collections. Note, that while the Dirichlet is a unimodal distribution, which assumes independence between the topics, a kernel density estimate using Dirichlet kernels can be multimodal and is able to capture dependencies between topics.

We propose to derive the unknown topic mixture $\vec{\theta}_q$ of an new document using a MAP estimator. The following quantities are given for the MAP estimator, namely the word vector $\vec{w}_q$ of the new document, the topic mixtures of the documents in the training set $\vec{\theta}$, and the word topic associations $\vec{\omega}$.



(a) (b) (c)

Figure 1. (a) topic mixtures (small icons) for documents learned by PLSI and for queries of different sizes (short: big empty icons, long: big filled icons) generated by PLSI folding-in, (b) typical likelihood in the topic simplex for a long query, (c) short query.

The maximum a posteriori estimator for the new document d_q is the topic mixture $\vec{\theta}_q^{MAP}$ which maximizes

$$P(\vec{\theta}_q|\vec{w}_q, \vec{\theta}, \vec{\omega}) \propto P(\vec{w}_q|\vec{\theta}_q, \vec{\theta}, \vec{\omega})P(\vec{\theta}_q|\vec{\theta}, \vec{\omega}). \quad (3)$$

where $P(\vec{w}_q|\vec{\theta}_q, \vec{\theta}, \vec{\omega})$ is the word likelihood and $P(\vec{\theta}_q|\vec{\theta}, \vec{\omega})$ is the topic mixture prior. We assume the query words to be independent, so the word likelihood can be decomposed as follows

$$P(\vec{w}_q|\vec{\theta}_q, \vec{\omega}) = \prod_{i=1}^{M}\sum_{j=1}^{K} P(w_i|a_j)P(a_j|q) = \prod_{i=1}^{M}\sum_{j=1}^{K} \omega_{ij}\theta_{qj} \quad (4)$$

Since the likelihood of the query words does not depend on $\vec{\theta}$, for ease of writing that parameter is neglected.

The prior is modeled by a kernel density estimate based on $\vec{\theta}$ using Dirichlet kernels. The topic mixtures are vectors, which have non-negative components and all components sum to one, $\forall 1 \leq l \leq N\colon \sum_{j=1}^{K} \theta_{lj} = 1$. That means those vectors reside in a $K-1$-simplex which is embedded in the $\mathbb{R}^K$. The Dirichlet distribution is suitable for such data since the density function integrates to one over the simplex. The density of a Dirichlet is given by

$$Dir(\vec{x}|\vec{\alpha}) = \frac{\Gamma(\sum_{j=1}^{K}\alpha_j)}{\prod_{j=1}^{K}\Gamma(\alpha_j)} \cdot \prod_{j=1}^{K} x_j^{\alpha_j - 1} \quad (5)$$

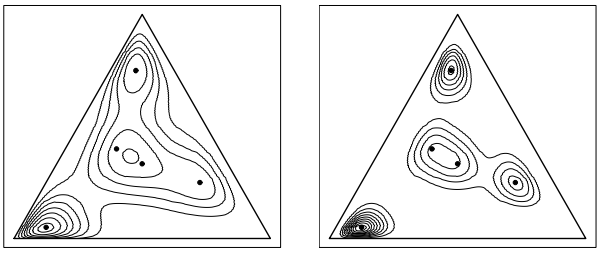
The jth coordinate of the mode of a Dirichlet is $\frac{\alpha_j - 1}{(\sum_{j'=1}^{K}\alpha_{j'}) - K}$ with $\alpha_j > 1$. The parameter vector $\vec{\alpha}$ controls both, the location of the mode in the simplex and the sharpness of the mode. Note, that multiplying the parameter vector with a scalar larger one means to increase the sharpness of the mode but it does not change the location of the mode.

A kernel density estimate sums over the given data, in our case the topic mixtures of the documents in the collection. The influence of each topic mixture θ_l is modeled by a single Dirichlet distribution, which has the mode located at θ_l. In order to control the sharpness of such a Dirichlet kernel the smoothing parameter h is introduced. Further, the function $\alpha(\vec{\theta}) = {}^1\!/_h \cdot \vec{\theta} + \vec{1}$ is introduced, which takes a topic mixture vector and outputs the corresponding parameter vector of the Dirichlet, s.t. the mode is exactly at $\vec{\theta}$. Large h makes the kernels flat and stretches the influences of the individual topic mixtures over the simplex, whereas small values for h make the kernels like sharp peaks. Modeling the prior as kernel density estimate based on the topic mixtures of the documents in the collection gives the following formular:

$$P(\vec{\theta}_q|\vec{\theta}) = \frac{1}{N}\sum_{l=1}^{N} \mathrm{Dir}(\vec{\theta}_q|\alpha(\vec{\theta}_l)) \quad (6)$$

Since the prior does not depend on $\vec{\omega}$, for ease of writing that parameter is neglected. Examples for priors in a $(3-1)$-simplex with different values for h are shown in figure 2.

The direct maximization of the righthand side of (3) with respect to $\vec{\theta}_q$ is difficult. Therefore, hidden variables are introduced for both, the word likelihood and the prior distri-



(a) (b)

Figure 2. Examples of density estimates with Dirichlet kernels, (a) $h = 0.07$, (b) $h = 0.02$.

bution to make the maximization tractable with an EM algorithm. First, the likelihood of a single word $P(w_i|\vec{\theta}_q, \vec{\omega})$ can be seen as a mixture model of K topics. Thus, a hidden binary variable $\vec{y}_i \in \{0,1\}^K$ is introduced, which indicates which topic a_j explains word w_i. Second, the prior $P(\vec{\theta}_q|\vec{\theta})$ (eq. 6) also can be seen as a mixture model with N Dirichlet components and equal component priors. Again, a hidden binary variable $\vec{z} \in \{0,1\}^N$ is introduced, which indicates the Dirichlet component which explains a specific setting of the topic mixture of the query document. All hidden variables are concatenated to the vectors $\vec{y}$ and $\vec{z}$ respectively. Instead maximizing the posterior shown in (3), the logarithm of the posterior of the extended model is maximized.

$$\begin{aligned}
&\log[P(\vec{\theta}_q|\vec{w}_q, \vec{y}, \vec{z}, \vec{\omega})] \\
&\quad = \log[P(\vec{w}_q, \vec{y}|\vec{\theta}_q, \vec{\omega}) P(\vec{\theta}_q, \vec{z}|\vec{\theta}, \vec{\omega})] \\
&\quad = \Big[\sum_{i=1}^{M}\sum_{j=1}^{K} y_{ij}\big[\log \omega_{ij} + \log \theta_{qj}\big]\Big] + \\
&\qquad \sum_{l=1}^{N} z_l\big[\log \frac{1}{N} + \log \mathrm{Dir}(\vec{\theta}_q|\vec{\alpha}(\vec{\theta}_l))\big] + c \qquad (7)
\end{aligned}$$

The constant c comes from the normalization constant in equation (3). The maximization is done by an EM-algorithm, which starts with some settings for the wanted topic mixture of the query document $\vec{\theta}^{(0)}$ and iteratively computes posteriors for the hidden variables in the E-step and updates the topic mixture of the query document in the M-step. The posteriors for the hidden variables computed in the E-step are given by the following formulas

$$P(y_{ij} = 1|w_i, \vec{\theta}_q^{(s)}, \vec{\omega}) = \frac{\omega_{ij} \cdot \theta_{qj}^{(s)}}{\sum_{j'=1}^{K} \omega_{ij'} \cdot \theta_{qj'}^{(s)}} = g_{ij} \qquad (8)$$

$$P(z_l = 1|\vec{\theta}_q^{(s)}, \vec{\theta}) = \frac{Dir\big(\vec{\theta}_q^{(s)}|\alpha(\vec{\theta}_l)\big)}{\sum_{l'=1}^{N} Dir\big(\vec{\theta}_q^{(s)}|\alpha(\vec{\theta}_{l'})\big)} = h_l \qquad (9)$$

In the M-step, the posteriors of the hidden variables are used as substitutes for the unknown values of the hidden variables and plugged into (7). That equation is maximized wrt. to $\vec{\theta}_q$ under the condition $\sum_{j=1}^{K} \theta_{qj} = 1$, which give the following update formula for the wanted topic mixture

$$\theta_{qj}^{(s+1)} = \frac{\sum_{i=1}^{M} g_{ij} + {}^1\!/_h \sum_{l=1}^{N} h_l \theta_{lj}}{M + {}^1\!/_h} \qquad (10)$$

Note that Bayesian folding-in includes PLSIs folding-in as a special case, namely when $h = \infty$. In that case, the Dirichlet kernels become flat and the posteriors h_l of the prior vanish in the update formula 10. Therefore, the posteriors of the prior become irrelevant and with $n(d_q) = M$ the formulas (8) and (10) reduce to the equations (1) and (2) of PLSI folding-in respectively.

Figure 3 continues the small example from the previous section and shows the results for Bayesian folding-in using the same data as before. The contour lines in figure 3(a) show the prior, which is the same for all queries. Also note, the multimodal posterior (figure 3c) for the short query (big empty triangle in the upper corner). The other less likely modes of that posterior may correspond to alternative semantic interpretations.

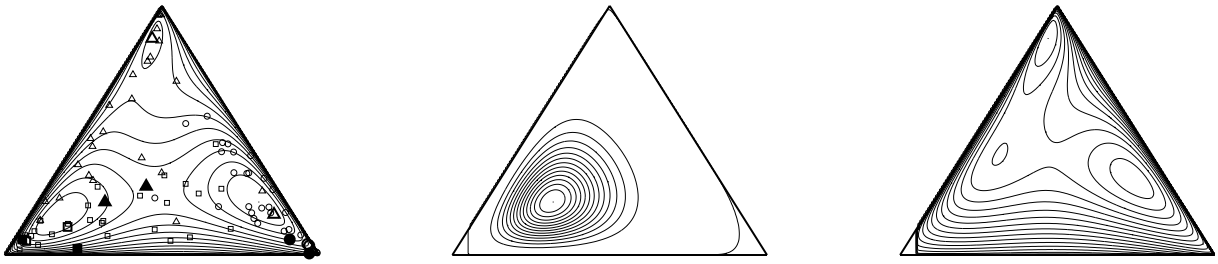
4. Bayesian Folding-In and Information Retrieval

An important application of Bayesian folding-in is information retrieval. A document collection is processed by PLSI and then queries are folded into the latent document space by Bayesian folding-in.

Bayesian folding-in determines for a query or a new document mixture proportions of latent topics, which can be seen as a K-dimensional vector. The found vector can be used to determine similarities to the documents of a given collection, for which such latent representations have been determined before. Usually the similarities are calculated by cosine similarity. Hofmann proposed model averaging [7], which linearly combines for a particular query document pair the similarities determined on different latent representations (for which usually the number of aspect varies) as well as the similarity determined on the original term representations.

Bayesian folding opens two new degrees of freedom to tune a ranking, namely (i) the choice of the starting point for folding-in and (ii) varying the smoothing parameter h.

Bayesian folding-in of a new document is a deterministic process, which starts with an initial topic mixture for that document and iteratively performs hill climbing on the posterior density eq. (3) until it converges towards a local maximum. However, as the posterior may have multiple local maxima it depends on the starting point of the hill climbing to which of the local maxima Bayesian folding-in converges in the end. From a data modeling perspective, the local maximum with the largest posterior, i.e. the global maximum, is preferred since we are doing maximum a posteriori estimation. The posterior consists of two factors, one which depends on the given new document or query at hand, i.e. called likelihood, while the other factor, called prior, is independent of the documents to be folded in. From practical experience the prior itself has typically multiple modes and is mainly responsible for the multimodal structure of the posterior. The likelihood has usually a single mode only. In order to find a small and general set of starting points, which is independent from the document to be folded in, the prior is analyzed only. Note, that the prior is independent of the document to be folded in, so that the following analysis has to be done only once as a preprocessing step.



(a) (b) (c)

Figure 3. (a) topic mixtures for documents learned by PLSI and for queries of different sizes (short, long) generated by Bayesian folding-in, (b) typical posterior in the topic simplex for a long query, (c) short query.

The idea to get the set of starting points is to determine the local maxima of the prior. This can be done by hill climbing as well. Note, that this maximization is a special case of the maximization of the posterior (3), just that the likelihood becomes a constant.

The answer to the choice of the starting point is a different one from the information retrieval perspective as from the data modeling perspective. For data modeling it is fine to select the start point, for which Bayesian folding-in converges to the global maximum of the posterior. However, information retrieval has the more informal goal to find documents relevant to the query. As especially short queries may have ambiguous meanings, the most appropriate representation of the query in the latent space may be not necessarily correspond to the global maximum of the posterior. Alternative meanings correspond to the set of local maxima of the posterior. Using relevance feedback, an information retrieval system may learn, which local maximum is most appropriate for the user and the retrieval task at hand. If no additional information (e.g. from relevance feedback) is available, the global maximum of the posterior gives the most plausible query representation.

The second new degree of freedom is the smoothing parameter h. The larger h the more the prior changes towards a flat distribution. The kernel density-based prior helps to focus onto the relevant part of the latent space during folding-in. A very large value for h effectively removes that focus and allows all possible mixture proportions of the latent topics for the query representation. In terms of information retrieval, the parameter h can be seen as a quantity, which specifies how general the query can be interpreted to match documents in the returned ranking. Large h means more general, since the latent query representation is allowed to take mixture proportions which are quite distant from the rest of the documents in the collection.

The discussion shows the potential of Bayesian folding-in for improvements in information retrieval.

5 Experiments

The performance of Bayesian folding-in is studied experimentally regarding the following question: does Bayesian folding-in improve the retrieval of relevant documents for a given query.

The lemur package[1] is used for preprocessing the text data as well as training PLSI on a collection of documents. Additionally the *trec_eval*[2] tool (version 8.1) is used to assist the evaluation of the information retrieval experiments.

The proposed applications of Bayesian folding-in are evaluated on public benchmark text corpora. The evaluation of Bayesian folding-in in combination with PLSI can be done on text documents only. For the application to information retrieval, queries with known relevant documents are additionally needed as ground truth to estimate precision and recall. Three document collections are used in this study, namely CISI (1460 document, 112 queries), CRAN (1400 documents, 225 queries) and MED (1030 documents, 30 queries) [3].

Preprocessing of documents and queries includes elimination of stop words using the list from the SMART project[4] as well as the elimination of infrequent words. Infrequent words are those that occur in less than δ documents and hence are assumed to obey only a minor information content about the topic mixture. In the experiment δ is chosen to be 5, which guarantees that no query becomes empty. Documents containing less than 5 words are neglected. All terms are reduced to word stems using Porters stemmer.

[1] `lemurproject.org`
[2] `trec.nist.gov/trec_eval`
[3] `ir.dcs.gla.ac.uk/resources/test_collections/`
[4] `http://ir.dcs.gla.ac.uk/resources/ir_sys`

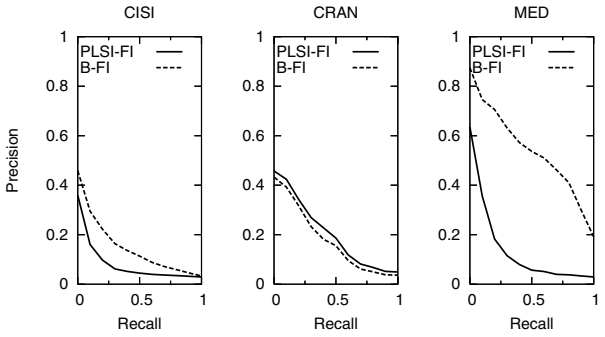


Figure 4. Interpolated recall-precision graphs of (i) PLSI folding-in, and (ii) Bayesian folding-in.

The experiment aims at assessing the capability of Bayesian folding-in in comparison to PLSI folding-in to be of use for information retrieval. In detail, the task is to retrieve a set of documents as the answers to a query with high precision and recall. For a given document collection and a query, recall is defined as:$|a \cap b|/|a|$ with the defined sets of relevant documents a and the retrieved documents b respectively. Whereas precision is defined as: $|a \cap b|/|b|$.

First PLSI is performed on the entire document collections for 32 topics resulting in a co-occurrence data model for each of them. As discussed in section 2 this model estimates for each document a vector in the latent topic space consisting of the learned document-topic mixtures. Afterwards, the queries are folded into the previously obtained co-occurrence data model using (i) PLSI folding-in, and (ii) Bayesian folding-in. The similarity between each query and all documents of the collection is computed and postprocessed to give interpolated recall-precision graphs. Similarities are defined as a weighted sum of similarities in the latent semantic space which are influenced by either PLSI-FI or B-FI, and in the original vector space spanned by the words. This strategy was proposed by Hofmann [7]. Since this study first aims at comparing PLSI-FI and B-FI the weight-parameter was not tuned.

The results of the experiment are shown in figure 4. In general, the more the graph approaches the upper right corner the better the retrieval performance. In case of CISI and MED one observes that for all recall values, the obtained precisions using Bayesian folding-in are above those using PLSI folding-in. In case of CRAN, the performances of both methods are comparable. These results indicate that (i) the B-FI benefit using a prior distribution over document-topic mixtures, and (ii) the B-FI is advantageous in information retrieval tasks.

The smoothing parameter $h = 0.02$ has been kept constant during all experiments. This value gives a reasonable small set of different starting points for Bayesian folding-in. In general, that parameter should be estimated, like other hyper-parameters of Bayesian methods, using cross-validation.

To conclude, a new Bayesian method for folding new documents into the latent space determined by PLSI is proposed. Additionally to PLSIs folding-in, a prior based on kernel density estimation with Dirichlet kernels is used. Bayesian folding-in has been applied to information retrieval and its superior performance in the scenario has been demonstrated on real document collections.

References

[1] D. Blei and M. Jordan. Variational inference for dirichlet process mixtures. *J. of Bayes. Anal.*, 1(1):121–144, 2005.

[2] D. Blei and J. Lafferty. Correlated topic models. *Advances in Neural Inf. Proc. Sys.*, 18, 2006.

[3] D. M. Blei and J. D. Lafferty. Dynamic topic models. In *ICML '06*, 113–120, 2006.

[4] D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent dirichlet allocation. *J. Mach. Learn. Res.*, 3:993–1022, 2003.

[5] P. V. Gehler, A. D. Holub, and M. Welling. The rate adapting poisson model for information retrieval and object recognition. In *ICML '06*, 337–344, 2006.

[6] M. Girolami and A. Kabán. On an equivalence between plsi and lda. In *SIGIR '03*, 433–434, 2003.

[7] T. Hofmann. Probabilistic latent semantic indexing. In *SIGIR '99*, 50–57, 1999.

[8] T. Hofmann. Learning the similarity of documents. *Advances in Neural Inf. Proc. Sys.*, 914–920, 2000.

[9] T. Hofmann. Unsupervised learning by probabilistic latent semantic analysis. *Machine Learning*, 42(1):177–196, 2001.

[10] X. Jin, Y. Zhou, and B. Mobasher. Web usage mining based on probabilistic latent semantic analysis. In *KDD '04*, 197–205, 2004.

[11] Q. Mei and C. Zhai. A mixture model for contextual text mining. In *KDD '06*, 649–655, 2006.

[12] M. Nyffenegger, J.-C. Chappelier, and Éric Gaussier. Revisiting fisher kernels for document similarities. In *ECML 2006*, 727–734, 2006.

[13] A. Popescul, L. H. Ungar, D. M. Pennock, and S. Lawrence. Probabilistic models for unified collaborative and content-based recommendation in sparse-data environments. In *UAI '01*, 437–444, 2001.

[14] M. Steyvers, P. Smyth, M. Rosen-Zvi, and T. Griffiths. Probabilistic author-topic models for information discovery. In *KDD '04*, 306–315, 2004.

[15] M. Welling, M. Rosen-Zvi, and G. Hinton. Exponential family harmoniums with an application to information retrieval. *Advances in Neural Inf. Proc. Sys.*, 17:1481–1488, 2005.