Finite Sequential Information Bottleneck (fsIB)

Jaakko Peltonen, Janne Sinkkonen, and Samuel Kaski
Neural Networks Research Centre
Helsinki University of Technology
P.O. Box 5400, FIN-02015 HUT, Finland
{jaakko.peltonen,janne.sinkkonen,sami.kaski}@hut.fi

Abstract

The sequential information bottleneck (sIB) algorithm clusters text documents or other data interpretable as co-occurring pairs of variables (analogous to documents vs. words). We introduce a variant of the algorithm that takes sparsity of data into account by postulating multinomial sources for the data. The new objective function, a Bayes factor, is asymptotically equivalent to mutual information, the original cost function of the sequential information bottleneck. Parameters of the multinomials need not be estimated, for they can be analytically marginalized away. Experiments imply performance equal to the original sIB for larger data sets and improved performance for clusters with small numbers of documents.

1 Introduction

In a common current approach to text document analysis, syntax is ignored and documents are treated as 'bags of words'. Under this model, the sufficient statistics for a document are the word counts n_y telling how many times each word y appears in the document.

At the level of a whole document collection, counts over documents x and all words y that appear within them form a matrix or, in statistical terms, a two-dimensional *contingency table*. Words of the document collection (columns) and the documents (rows) form the margins of this table. Formally, we consider a set of co-occurrences of two categorical variables X and Y. (Values of the random variables are denoted by lowercase letters.) The co-occurrences can be enumerated one by one, but given exchangeability, a sufficient statistic for the set of co-occurrences are the counts n_{xy} of various combinations (X, Y). These form a two-dimensional contingency table.

Clustering the document collection is equivalent to collecting together rows of the corresponding contingency table. More generally, any data expressible as co-occurrences between two discrete variables is statistically equivalent to the document collection under the bags of words model. Similar statistical methods then apply, including those presented in this paper.

Methods for clustering contingency table margins have been proposed earlier both in statistics [4, 6] and in machine learning [1, 2, 7, 8, 11, 14]. We focus on the sequential IB algorithm (sIB; [12]), an application of the information bottleneck principle [14]. IB and sIB try to preserve the structure of the contingency table in the sense of making the clusters maximally predictive or dependent about the other margin of the table. Dependency is measured by mutual information.

IB and sIB are based on known probability distributions instead of counts. Theoretically this implies that they are asymptotic algorithms, and that their accuracy for finite, especially sparse data sets, is uncertain. We propose a theoretically justified cost function defined for counts, and an algorithm very similar to sIB. For large data sets the new cost function approaches mutual information, and the algorithm approaches sIB. Empirical performance is comparable to sIB and better for very sparse data.

The current work derives from our previous work on clustering algorithms for continuous data [10]; In the present paper the cost function is presented as a Bayes factor, which is its most general interpretation and directly applicable to the case of two adjustable margins. In the present one-margin case, a likelihood interpretation is also available. The likelihood is a marginalized version of the crisp-cluster likelihood interpretation of IB ([13], but see also the ACM model in [8]).

1.1 Information bottleneck

The information bottleneck (IB) principle for clustering rows of contingency tables is based on the Shannon's information theory and is therefore asymptotic: Instead of counts n_{xy}, it deals with the joint distribution $p(x, y)$ of X and Y. Of course, the joint distribution is interpretable as the expected relative counts $n_{xy}/\sum_{x'y'} n_{x'y'}$ when the total number of co-occurrences $\sum_{xy} n_{xy}$ is large. The IB principle and algorithms that follow are therefore usable for the analysis of real data sets as well, although straightforward applications of the principle may suffer from the excess noise in finite counts computed from sparse data.

In information-theoretic terms, IB finds a representation T for X. Technically, T is a discrete random variable. The representation can be identified with the conditional distribution $p(t|x)$ of clusters t given the value x of the margin variable X.

From the viewpoint of clustering, t can be interpreted as clusters. Cluster assignments are then not deterministic but the indeterminacy, the probabilities $p(t|x)$, can be interpreted as degrees of cluster memberships.

Objective function. IB minimizes the cost $I(X, T) - \beta I(T, Y)$, which is a compromise between two mutual informations. The dependency term $I(T, Y)$, mutual information between Y and T, tries to preserve the structure of the joint distribution $p(X, Y)$, while the complexity term $I(X, T)$ tries to keep the representation T non-informative of X, or simple. The representation T can be seen as a bottleneck squeezing out essential parts of the dependency between X and Y.

A compromise that depends on β is made, and one of the original motivations behind the IB setup is to study the properties of the continuum spanned by β. Variational optimization of the cost with respect to $p(t|x)$ leads to a solution characterized by the three equations

$$\begin{cases} p(t|x) = \frac{p(t)}{Z(\beta,x)} \exp(-\beta D_{KL}(p(y|x), p(y|t))) \ , \\ p(y|t) = \frac{1}{p(t)} \sum_x p(y|x)p(t|x)p(x) \ , \\ p(t) = \sum_x p(t|x)p(x) \ . \end{cases} \tag{1}$$

Of these equations only the first is of particular interest, for it characterizes the shape of the optimal clusters spanned over the document space X. Before normalization by $Z(\beta, x) = \sum p(t) \exp(\cdot)$, the clusters are exponentially decaying functions of the Kullback-Leibler divergence from a document 'prototype' $p(y|t)$. (Note that it suffices to characterize the documents by their word distributions $p(y|x)$.)

The equations (1) propose an alternating, iterative optimization algorithm for finding clusters $p(t|x)$, given $p(x,y)$ and β. The algorithm resembles the Blahut-Arimoto algorithm of rate distortion theory [3].

Crisp clusters. From (1) it is clear that as $\beta \to \infty$, optimal clusters $p(t|x)$ tend to become crisp or sharp, in the sense that $p(t|x) \to 0$ or $p(t|x) \to 1$ almost everywhere. It is even more clear from the cost function $I(X,T) - \beta I(T,Y)$ that for $\beta \to \infty$ it essentially becomes $I(T,Y)$, a measure of dependency between T and Y. In other words, when the dependency $\beta I(T,Y) \propto I(T,Y)$ is variationally maximized with respect to the cluster shapes $p(t|x)$, we find that the optimal clusters are necessarily of the form $p(t|x) \in \{0,1\}$.

If the main motivation of a data analysis setup is to group the data, it is tempting to at least finally set $\beta \to \infty$. This is the cluster shape of the sequential information bottleneck (sIB) algorithm, and also the case on which we concentrate in this paper.

In the case of crisp clusters, representation T becomes a deterministic function $t(x) : X \to T$ of the original data x. Just as the co-occurrences of X and Y form a contingency table, the co-occurrences of T and Y then form another, smaller co-occurrence table. The (T,Y) table is obtained from the (X,Y) table by collating rows together according to $t(x)$. Below we will denote the (T,Y) table by $\mathbf{N}$ and its entries by n_{ty}. Margins of the table are denoted by $n_t = \sum_y n_{ty}$, $n_y = \sum_t n_{ty}$, and the total number of co-occurrences in the table by $N = \sum_{t,y} n_{ty}$.

Note that for soft clusters, the contingency table between T and Y does not exist, because assignments at the T-margin would not be well defined.

Although the setting is more general, for convenience x, y, and t are below called documents, words, and clusters, respectively.

1.2 Sequential information bottleneck (sIB)

The sequential information bottleneck method (here sIB for brevity; [12]) produces crisp clusters, $p(t|x) \in \{0,1\}$, with each document belonging to one cluster only. This corresponds to the maximizing the dependency of the clusters T and the words Y, measured by the mutual information $I(T,Y)$. It may therefore be seen as a margin-collating contingency table algorithm in the sense explained above.

Given an initial assignment of documents x to clusters t, the sIB algorithm sequentially draws a random document x from the solution, and finds a new cluster assignment for it by minimizing a merging criterion. In the sIB algorithm the merging criterion is (weighted) Jensen-Shannon divergence between the word distributions of the relocated document and the candidate cluster, i.e.,

$$\begin{aligned}(p(x)+p(t)) &\cdot D_{JS}(p(Y|x), p(Y|t)|\pi_1, \pi_2) \\ &= (p(x)+p(t)) \cdot [\pi_1 D_{KL}(p(Y|x), \bar{p}(Y)) + \pi_2 D_{KL}(p(Y|t), \bar{p}(Y))] \qquad (2)\end{aligned}$$

where D_{KL} is the Kullback-Leibler divergence, $\pi_1 = p(x)/(p(x)+p(t))$, $\pi_2 = p(t)/(p(x)+p(t))$ and $\bar{p}(y) = \pi_1 p(y|x) + \pi_2 p(y|t)$.

In experiments sIB outperformed agglomerative IB, iterative double clustering, K-means and a number of heuristic alternatives for sequential clustering [12]. The sequential approach may also perform better than alternating optimization by equations (1) [12].

2 Sequential information bottleneck (sIB) for finite data

Mutual information is an asymptotic quantity, defined for document and word probabilities $p(x)$ and $p(y|x)$. The implicit solution (1) of the IB optimization problem therefore also refers to probability distributions. In straightforward data analysis tasks the probabilities are unknown; for IB they need to be estimated from a data set. In [12] the estimate was $p(y|x) \propto n_{xy}$, where n_{xy} is the number of co-occurrences of word y in document x; this is a maximum likelihood estimate.

With sparse data, however, the estimates are inaccurate and point estimates, including ML, can lead to overfitting. Word counts of short documents are noisy descriptors of the topical content of the documents, but the noise is not visible in the ML (point) estimates.

A solution is to replace the point estimates by the posterior distributions of document content, given the counts and the bag-of-words model. Uncertainty is integrated out only at the final stage of evaluating the evidence for the model. Then uncertainty induced by finite word counts gets propagated to a higher level of the analysis, and uncertain data are correctly given less weight. Assuming the bag-of-words model is sound, such an analysis should better match the underlying topical categories, and improve generalization to new data.

In this section we present a finite-data variant of the Sequential Information Bottleneck method, called *finite sequential information bottleneck* (fsIB), that takes the finite length of documents into account in a probabilistic manner. The novelty in the variant is contained in the new objective function; otherwise the resulting algorithm is identical to sequential IB with a similar time complexity and similar convergence proofs.

We will first derive the objective by considering generative models and their Bayes factors; we then show that at the limit of large co-occurrence counts the objective is equivalent to mutual information.

2.1 Objective function

We aim to measure the dependency between the rows and the columns of the table. A Bayesian measure is the Bayes factor between two model families [5]. The first family H_1 models the entire contingency table by a single multinomial distribution with a Dirichlet prior. The second family H_2 models the marginals (clusters and words) by independent multinomial distributions with Dirichlet priors. That is, the first family assumes the clusters may depend on the words and the second assumes them independent. Both families are generative for the words but not for the documents.

The objective function C_{fsIB} of the fsIB method is (up to a constant factor) the

Bayes factor comparing the two model families. It can be shown (Appendix A) that this is given by

$$\frac{p(\mathbf{N}|H_1)}{p(\mathbf{N}|H_2)} \propto \frac{\prod_{t,y}\Gamma(n_{ty}+\alpha_{ty})}{\prod_t \Gamma(n_t+\alpha_t)} = C_{fsIB} \tag{3}$$

where the α_{ty} are parameters of the Dirichlet prior for the co-occurrence counts and the α_t are parameters of the Dirichlet prior for the sizes of the clusters. The Dirichlet parameters α_{ty} and α_t can take any positive values—note that in general they do not sum up to unity.

The Bayes factor compares the likelihood of the data given the two families, H_1 in the numerator and H_2 in the denominator. For both families, the Bayes factor accounts for the uncertainty about a document's generating word distribution by integration over its parameters. This integration is done collectively for all documents in each cluster, since they are assumed to be generated by the same distribution.

A simple interpretation. A special interpretation of the Bayes factor is possible if "consistent priors" are used for the model families, that is, if $\alpha_t = \sum_y \alpha_{ty}$. In this special case the Bayes factor is inverse to the *evidence term* in the posterior probability of independent margins (Appendix B). That is, $p(H_2|H_1, \mathbf{N}) \propto p(H_2|H_1)/C_{fsIB}$. The posterior is computed given the data $\mathbf{N}$ and the (larger) model family H_1. Maximizing the Bayes factor minimizes the evidence that the margins are independent.

Interpretation as marginalized likelihood. When consistent priors are used, it turns out the objective function (3) can also be interpreted as the marginal likelihood $p(\mathbf{N}|H_3)$ under a family H_3 that models each cluster with a separate multinomial distribution, with a Dirichlet prior α_{ty} for each cluster t [10]. The likelihood is marginalized over the individual models (parameter values) in H_3. (Compared to [10], the novelty here is the application to discrete data.)

Connection to the hypergeometric formula. Also note that if $\alpha_{ty} = \alpha_t = 1$, the gamma functions in (3) turn to factorials, and the cost function becomes a simple counting argument resembling those behind the concept of physical entropy.

2.2 Connection to sequential IB

On document priors. The equations (1) are defined for any distributions $p(x)$ and $p(y|x)$. In [12], the sequential IB was tested using a uniform prior over documents, $p(x) = 1/|X|$, where $|X|$ is the number of documents in the dataset. On the other hand, emphasizing long documents as more informative may be desirable. Then the non-uniform document prior proportional to the number of words in the documents, $p(x) = n_x/N$, is an obvious candidate (n_d is the number of word occurrences in the document and N is the total number of word occurrences in the dataset, and hence in the contingency table).

Asymptotically equivalent objectives. It turns out that in the limit of large co-occurrence counts, i.e. $N \to \infty$, the fsIB objective function C_{fsIB} is equivalent (Appendix C) to the sequential IB objective $I(T,Y)$, when $I(T,Y)$ is computed based on the non-uniform document prior.

For a uniform document prior such a connection is not known, however (outside the trivial case of equal-length documents). Both priors are possible choices for sequential IB. The existence of the Bayes factor connection argues in favor of the nonuniform prior, but it is an empirical issue which prior is better in practice. In Section 3 we test the sequential IB method with both priors.

Other connections. In addition to the information bottleneck method, the proposed model can also be related to the asymmetric clustering model (ACM; [8]). The difference is that we integrate over the word distribution parameters, instead of using a maximum likelihood point estimate.

2.3 Optimization

In [12] a 'template algorithm' for sequential clustering algorithm was introduced. It applies to optimizing so-called *decomposable* objective functions, i.e., sums over functions that depend only on an individual cluster. Applying this template for mutual information yielded the sequential Information Bottleneck algorithm, and various heuristic choices were also tested.

The same template algorithm can be applied to optimizing the fsIB objective function (3), since its logarithm is decomposable. The resulting algorithm resembles the sequential Information Bottleneck; the only changes are that the objective function and similarity measure are replaced by their fsIB equivalents.

Cluster merging criterion. For sequential IB, the new cluster for an extracted document was chosen by minimizing a weighted Jensen-Shannon divergence. For fsIB, the criterion is to maximize the Bayes factor (3).

If a document $\mathbf{d}$ is extracted from some cluster t_0 and merged to cluster t, the Bayes factor is changed by a factor of

$$d_{fsIB}(\mathbf{d}, t) = \left(\prod_y \frac{\Gamma(n_{ty} + d_y + \alpha_{ty})}{\Gamma(n_{ty} + \alpha_{ty})} \right) \cdot \frac{\Gamma(n_t + \alpha_t)}{\Gamma(n_t + |\mathbf{d}| + \alpha_t)} \tag{4}$$

times a constant. Here n_{ty} and n_t are the co-occurrence counts after the document has been extracted from t_0, d_y is the co-occurrence count of word y in the document and $|\mathbf{d}| = \sum_y d_y$. Note that this quantity depends only on the co-occurrences in the document and the cluster t, not on the rest of the contingency table. The choice t that yields the largest value of the criterion is chosen.

Interpretation of the criterion. It can be shown (see Appendix D) that the cluster merging criterion (4) is asymptotically equal to the weighted Jensen-Shannon divergence (2) used in sequential IB.

Notice that when $n_{t_0 y}$ and n_{t_0} have been updated after extracting the document, the criterion (4) does not depend further on the original cluster t_0. In fact, adding a completely new document to the contingency table yields the same criterion. This yields probabilistic interpretations for the criterion.

In general, if new data $\mathbf{N}_2$ is added to the contingency table, the change in the

Bayes factor is given by

$$\frac{p(\mathbf{N}, \mathbf{N}_2|H_1)}{p(\mathbf{N}, \mathbf{N}_2|H_2)} \cdot \frac{p(\mathbf{N}|H_2)}{p(\mathbf{N}|H_1)} = \frac{p(\mathbf{N}_2|H_1, \mathbf{N})}{p(\mathbf{N}_2|H_2, \mathbf{N})} \tag{5}$$

i.e. it is simply a Bayes factor comparing H_1 and H_2 with data $\mathbf{N}_2$ after witnessing $\mathbf{N}$. For the fsIB algorithm the 'new data' is simply the previously extracted document, i.e., $\mathbf{N}_2 = \mathbf{d}$.

Furthermore, if consistent priors are used, the criterion (4) is inverse to the *new evidence* term for updating the Bayesian posterior probability of independent margins. That is, we have

$$p(H_2|H_1, \mathbf{N}, \mathbf{N}_2) = \frac{p(\mathbf{N}_2|H_1, H_2, \mathbf{N})}{p(\mathbf{N}_2|H_1, \mathbf{N})} \cdot p(H_2|H_1, \mathbf{N}) \tag{6}$$

where the first term on the right is (inverse to) the cluster merging criterion when $H_2 \subset H_1$, and the second term is independent of $\mathbf{N}_2$. Notice that again the first term contains all the factors that depend on the new data; therefore it can be called the *new evidence* term. Minimizing it keeps the evidence of independent margins minimal.

In the case of consistent priors, there is a third probabilistic interpretation: it turns out that (4) is equal (up to a constant factor) to $p(\mathbf{d}|\mathbf{n}_t)$, the probability that the (unknown) multinomial distribution that generated the words $\mathbf{n}_t$ in cluster t also generated the words in $\mathbf{d}$, when a Dirichlet prior with parameters α_{ty} is used for the multinomial distribution.

The algorithm. For completeness, we present pseudo-code for the fsIB optimization algorithm in Fig. 1. The algorithm takes n different restarts from random clusterings into K clusters. For each restart, the algorithm continues until it has performed $maxL$ iterations over the data set, or until the number of cluster changes per iteration is at most a fraction ϵ of the number of objects.

Convergence and complexity. The convergence result for the sequential IB algorithm also holds for the fsIB algorithm; since cluster changes are directly chosen to maximize the objective function, it never decreases during the iteration. 'Loops' over equally good clusterings are not possible since for each document, the highest cluster index with the largest objective function value is chosen, and therefore they cannot move back to a previous cluster with equal objective function value. Moreover, the fsIB objective is upper bounded for any dataset as long as α_{ty} and α_t are positive, since the gamma functions are then bounded above and below. The empirical priors used in Section 3 satisfy this requirement.

Since sequential IB and fsIB use essentially the same optimization method with different cost functions (which are equally complex to evaluate), both methods have the same time complexity $O(K|Y|)$ per iteration. The total time complexity is bounded by $O(nLK|X||Y|)$ for both methods, where n is the number of restarts, L is the number of cycles over X and K is the number of clusters. In practice the fsIB algorithm seems to converge with less iterations than sequential IB.

Input:
$|X|$ objects to be clustered
Parameters: K, n, $maxL$, ϵ

Output:
A partition T of X into K clusters

Main Loop:
For $i = 1, \ldots, n$
    $T_i \leftarrow$ random partition of X.
    $c \leftarrow 0$, $C \leftarrow 0$, $done = FALSE$
    While not *done*
        For $j = 1, \ldots, |X|$
            Draw x_j out of $t(x_j)$
            $t^{new}(x_j) = \arg\min_{t'} d_{fsIB}(\{x_j\}, t')$
            If $t^{new}(x_j) \neq t(x_j)$ then $c \leftarrow c + 1$
            Merge x_j into $t^{new}(x_j)$
        $C \leftarrow C + 1$
        If $C \geq maxL$ or $c \leq \epsilon \cdot |X|$ then
            $done \leftarrow TRUE$
$T \leftarrow \arg\max_{T_i} C_{fsIB}(T_i)$

Figure 1: Pseudo-code for the fsIB optimization algorithm. In the context of clustering text documents, the x_j are documents with word co-occurrences $\{x_j\}$, and $t(x_j)$ and $t^{new}(x_j)$ are the old and new clusters of extracted document x_j. The overall criterion C_{fsIB} is computed by (3) and the cluster merging criterion d_{fsIB} by (4).

3 Empirical tests

In this section we carry out a set of experiments to verify that the fsIB yields improved unsupervised clustering results. Since the sequential IB (sIB) algorithm has already been compared to several alternatives in [12], and outperformed them, we will compare fsIB only to sIB.

3.1 The data sets

We compared fsIB and sequential IB on the same two datasets used in [12]. The Twenty Newsgroups (20NG; [9]) and Reuters-21578 (Reuters; available from http://www.daviddlewis.com/resources/testcollections/reuters21578/) datasets are standard test corpuses of text documents. The 20NG set consists of articles from 20 Usenet newsgroups, and the Reuters set consists of short news articles from the Reuters stream.

In both datasets, a categorization of the documents is known; for the 20NG set the newsgroups are the categories. For the Reuters set news articles belong in one or more topic categories. For both datasets, the number of clusters sought equaled the number of categories. The categories themselves were not used in training, but were only used to evaluate the clustering results.

In [12] the methods were also compared on a selection of subproblems from the two corpuses; here we only tested the full problems.

Preprocessing. In most details we replicated the preprocessing made in [12]. For the 20NG set the preprocessing included removing stopwords, setting characters to lowercase, replacing digits with zeroes, ignoring non-alphanumeric characters and pruning words that only occurred once. For the Reuters set a similar but slightly more complicated preprocessing was performed, including more advanced word splitting, and replacement of numeric expressions with indicator tags.

After the initial preprocessing the 2000 most informative words were selected, i.e. the words with the largest terms in the equation of mutual information between documents and words [12], that is, the words y with largest values of

$$\sum_x p(x,y) \log \frac{p(x,y)}{p(x)p(y)} \ . \tag{7}$$

Note that the 'informativeness' is based on a joint distribution estimate $p(x,y)$ which could be computed with either a uniform or non-uniform document prior,as described in Section 2.2. Here we used the non-uniform prior. Lastly, documents that had less than 10 words were discarded. This yielded 18656 documents for the 20NG set and 8452 for the Reuters set.

3.2 Test setup

Comparison methods. The fsIB algorithm is compared against the sIB algorithm, with two variants for both methods. For sequential IB, both uniform (sIB 1) and nonuniform (sIB 2) document priors are used.

For fsIB, two kinds of priors are used for the generative model parameters. Both are empirical priors with $\alpha_{ty} = n_y \frac{|Y|}{N}$, where $|Y|$ is the number of possible values of Y (size of the dictionary) and N is the total number of co-occurrences. The priors differ for α_t: we set either $\alpha_t = 1$ or $\alpha_t = |Y|$. The first choice (fsIB 1) leaves the priors α_{ty} and α_t inconsistent, and the second (fsIB 2) makes them consistent.

For all variants we set the parameters of the optimization algorithm to $maxL = 30$, $\epsilon = 0$ and $n = 15$.

Data subdivision The fsIB algorithm is expected to be especially effective for sparse data; for non-sparse data it approaches sequential IB with a nonuniform document prior. To test the effect of sparseness, both datasets were divided into an increasing number of subsets (from 1 to 160) by stratified sampling; with a fixed number of clusters, this leaves progressively less samples per cluster.

For each subdivision of a dataset (20NG or Reuters), all subsets were first clustered by all methods. The performances of the methods were then compared across the subsets by paired t-test. Note that too low numbers of subsets will yield large p-values; on the other hand, too small subsets may cause noise in the results.

Goodness measure The goodness of the results is evaluated by their micro-averaged precision with respect to human-given categories: the newsgroup of each

Dataset	subsets	clusters	doc./ cluster	sIB 1	sIB 2	fsIB 1	fsIB 2
20NG	1	20	932.8	**56.6**	50.6	52.4	51.9
20NG	10	20	93.3	44.1	$\underline{\underline{47.5}}$	**48.9**	48.4
20NG	20	20	46.6	40.1	$\underline{\underline{44.0}}$	45.1	**45.4**
20NG	40	20	23.3	38.3	40.9	$\underline{\underline{43.5}}$	$\underline{\underline{\mathbf{44.7}}}$
20NG	80	20	11.7	36.6	38.1	38.8	$\underline{\underline{\mathbf{42.1}}}$
20NG	160	20	5.8	$\underline{\underline{38.9}}$	38.9	37.6	$\underline{\underline{\mathbf{43.9}}}$
Reuters	1	10	845.2	85.2	88.5	**89.3**	88.4
Reuters	10	10	84.5	84.9	83.5	82.5	**85.5**
Reuters	20	10	42.3	**84.4**	83.3	82.7	84.0
Reuters	40	10	21.1	82.3	82.2	79.8	**83.4**
Reuters	80	10	10.6	$\underline{81.3}$	$\underline{\underline{79.9}}$	77.3	$\underline{\mathbf{82.6}}$
Reuters	160	10	5.3	$\underline{\underline{81.2}}$	$\underline{\underline{79.9}}$	72.0	$\underline{\underline{\mathbf{83.4}}}$

Table 1: Class discovery accuracy of the different methods, measured as micro-averaged precision (8). sIB 1: sIB with uniform document prior, sIB 2: sIB with normalized document length as document prior, fsIB 1: fsIB with empirical word prior, total weight over words = number of dimensions, inconsistent cluster prior with total weight = number of clusters, fsIB 2: fsIB, empirical word prior whose total weight = number of dimensions, consistent cluster prior. Numbers are average micro-averaged precisions over the subsets, in percentages. The best results are in boldface. If a number is underlined, $p < 0.05$ for a two-tailed paired t-test between it and the next best method, and if the number is doubly underlined, $p < 0.01$.

post for the 20 Newsgroups set and the topic categories of the news article for the Reuters set. In the Reuters set each document may belong to several categories.

Assume first that for a clustering T, each cluster is given the class label of the dominant class in the cluster, i.e., by majority vote of documents in the cluster. The clustering classifies each document to the label of its cluster. Given this classification, the micro-averaged precision of the clustering T is defined by

$$P(T) = \frac{\sum_c f_1(c,T)}{\sum_c f_1(c,T) + f_2(c,T)} = \frac{1}{|X|} \sum_c f_1(c,T) \tag{8}$$

where $f_1(c,T)$ is the number of documents correctly classified to class c (their true classes include c), $f_2(c,T)$ is the number of documents incorrectly classified to c and $|X|$ is the total number of documents.

In [12] a thresholding strategy was introduced to increase precision at the expense of recall. Here we only compare precision, but the same thresholding strategy can be used for fsIB as well.

3.3 Results

Table 1 presents the micro-averaged precision results, averaged over subsets. Figure 2 presents the average results over the 20NG and Reuters sets as curves.

Note that the results for sequential IB on the whole data are very close to those in [12]; the differences in preprocessing are then likely to be minor.

As expected, performance decreases for all methods as the data becomes sparse

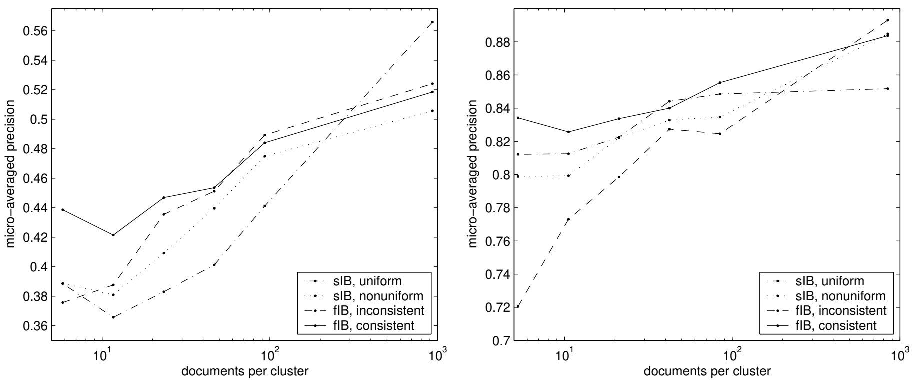

Figure 2: Micro-averaged precision results, averaged over subsets. Left: Twenty Newsgroups data, right: Reuters data.

(documents/cluster becomes small). However, the fsIB algorithm with consistent priors (fsIB 2) yields the best results on the sparse data. As the amount of data increases the fsIB results converge to sIB with the nonuniform prior (sIB 2), as expected based on the asymptotical equivalence of their objective functions.

For fsIB, the consistent priors yield clearly better results for on the small data sets; the non-consistent priors seem slightly better on the largest sets but the differences are not significant.

For sequential IB, neither of the document priors is clearly better: a uniform prior is better on the small Reuters subsets, but worse on the whole set. The non-uniform prior is better on the small 20NG subsets, but worse on the whole set. The last difference is large enough that the fsIB variants, asymptotically equivalent to the non-uniform prior, also perform poorly.

It is surprising that the choice of document prior seems to depend on the size of the dataset; the significance of the whole-set results is not known, though, and the algorithms may e.g. be caught in local minima. Still, the result merits more investigation.

Why are sIB 1 and the fsIB versions bad on the 20NG set in the large data limit? All three algorithms (asymptotically) weigh documents by their length. We tested the effect of removing the longest and shortest five percent of documents, but this did not improve sIB 1 performance on the whole set. However, the 20NG set contains some leftover words from the subject lines; such words do not follow the bag of words model (a longer article would not have more subject lines), so it is tentatively possible they worsen performance of weighting based on the bag of words. More investigation would be needed to verify the hypothesis.

Why is fsIB 1 bad on the Reuters set in the small data limit? The inconsistent prior in fsIB 1 effectively pays more attention to cluster sizes and requires more uniform cluster sizes. If the underlying topics (categories) have very non-uniform prior probabilities, requiring uniform cluster sizes prevents the algorithm from finding the categories. The effect is strongest on small datasets where the influence of the prior is largest. From the results we found that on the whole dataset fsIB 1 and fsIB 2 had found clusters of very similar sizes, while on the smallest subsets the fsIB 1 clusters were much more uniform-sized than the fsIB 2 clusters. Because the actual class sizes of the Reuters dataset are highly non-uniform this clearly yields

Dataset	subsets	clusters	doc./ cluster	sIB 1	sIB 2	fsIB 1	fsIB 2
20NG	1	20	932.8	30.0	28.0	27.0	28.7
20NG	10	20	93.3	23.2	18.4	19.0	17.7
20NG	20	20	46.6	19.0	16.0	15.7	14.5
20NG	40	20	23.3	14.4	13.0	13.8	11.5
20NG	80	20	11.7	10.9	10.2	11.4	9.0
20NG	160	20	5.8	8.1	7.9	8.1	6.7
Reuters	1	10	845.2	18.9	18	17.9	17.2
Reuters	10	10	84.5	11.8	10.4	11.5	10.0
Reuters	20	10	42.3	9.5	9.2	9.8	8.6
Reuters	40	10	21.1	8.2	7.6	9.2	6.9
Reuters	80	10	10.6	6.7	6.4	8.2	5.6
Reuters	160	10	5.3	5.4	5.1	6.1	4.6

Table 2: Convergence speed measured by number of iterations over the data, averaged over restarts and subsets. sIB 1: sIB with uniform document prior, sIB 2: sIB with normalized document length as document prior, fsIB 1: fsIB with empirical word prior, total weight over words = number of dimensions, inconsistent cluster prior with total weight = number of clusters, fsIB 2: fsIB, empirical word prior whose total weight = number of dimensions, consistent cluster prior.

suboptimal results.

We additionally made initial tests on INSPEC data (divided to 70 subsets). The INSPEC data are abstracts of scientific documents with classes given by informaticians. However, in the initial tests all four methods performed about equally poorly: the micro-averaged precision was about 24.2 percent for all methods, differences being below 0.1 percent.

3.4 Convergence speed

Table 2 presents results of the convergence speed of the algorithms, in average numbers of iterations over the respective datasets. For all algorithms, the convergence takes longer for the larger sets.

Of the two IB versions, the version with uniform document prior takes on average slightly longer to converge. The fsIB algorithm with consistent priors (fsIB 2) on average converges in the smallest number of iterations, except on the whole 20NG set. However, since the (log) Gamma function in fsIB is slower to compute than the logarithm in sIB, fsIB still takes more actual time to run than sIB. It may be worthwhile to consider faster approximations to the Gamma function to gain a speed advantage.

4 Conclusions and discussion

The finite sequential information bottleneck (fsIB) algorithm gives a probabilistic Bayes Factor interpretation to distributional crisp clustering by the sequential Information Bottleneck method. Asymptotically, the Bayes factor objective function becomes equivalent to mutual information. Moreover, a second interpretation as an evidence term exists when consistent priors are used. In addition, fsIB inherits desirable properties such as convergence, computational complexity and a thresholding strategy from sequential IB.

In empirical tests, fsIB yields improved clustering performance on small (sparse) data sets, and on large sets it performs equally to sequential IB with a nonuniform document prior; which prior is better for large data sets remains an open question.

Consistent priors were essential to the fsIB performance. In this paper the prior strength was fixed, but a prior specific to the dataset (chosen e.g. with cross-validation) could further improve performance.

In this paper we consider only one-sided clustering, but the Bayes factor objective function has been used for two-sided clustering as well.

References

[1] D. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[2] W. Buntine. Variational extensions to EM and multinomial PCA. In T. Elomaa, H. Mannila, and H. Toivonen, editors, *Proceedings of the ECML'02, 13th European Conference on Machine Learning*, Lecture Notes in Artificial Intelligence 2430, pages 23–34. Springer, Berlin, 2002.

[3] Thomas M. Cover and Joy A. Thomas. *Elements of Information Theory.* Wiley, New York, 1991.

[4] Zvi Gilula. Grouping and association in contingency tables: An exploratory canonical correlation approach. *Journal of the American Statistical Association*, 81:773–779, 1986.

[5] I. J. Good. On the application of symmetric Dirichlet distributions and their mixtures to contingency tables. *Annals of Statistics*, 4(6):1159–1189, 1976.

[6] L. A. Goodman. Criteria for determining whether certain categories in a cross-classification table should be combined with special reference to occupational categories in an occupational mobility table. *American Journal of Sociology*, 87:612–650, 1981.

[7] Thomas Hofmann. Probabilistic latent semantic analysis. In *Proceedings of the Fifteenth Conference on Uncertainty in Articial Intelligence*, pages 289–296. Morgan Kaufmann Publishers, San Francisco, CA, 1999.

[8] Thomas Hofmann and Jan Puzicha. Statistical models for co-occurrence data. A.I. Memo 1625, MIT, 1998.

[9] Ken Lang. NewsWeeder: Learning to filter netnews. In *Proceedings of the 12th International Conference on Machine Learning*, pages 331–339. Morgan Kaufmann Publishers, San Mateo, CA, 1995.

[10] Janne Sinkkonen, Samuel Kaski, and Janne Nikkilä. Discriminative clustering: Optimal contingency tables by learning metrics. In T. Elomaa, H. Mannila, and H. Toivonen, editors, *Proceedings of the ECML'02, 13th European Conference on Machine Learning*, pages 418–430. Springer, Berlin, 2002.

[11] Noam Slonim. *The information bottleneck: theory and applications.* PhD thesis, Hebrew University, Jerusalem, 2002.

[12] Noam Slonim, Nir Friedman, and Naftali Tishby. Unsupervised document classification using sequential information maximization. In *Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 129–136. ACM Press, 2002.

[13] Noam Slonim and Yair Weiss. Maximum likelihood and the information bottleneck. In *Advances in Neural Information Processing Systems 14*. MIT Press, Cambridge, MA, 2002.

[14] Naftali Tishby, Fernando C. Pereira, and William Bialek. The information bottleneck method. In *37th Annual Allerton Conference on Communication, Control, and Computing*, pages 368–377. Urbana, Illinois, 1999.

A Derivation of the fsIB objective function

The integral of a probability distribution is 1. Therefore, if the distribution is of the form $\frac{1}{C} \cdot f(\theta)$, the integral of $f(\theta)$ is C. In particular, for a Dirichlet distribution

$$p(\theta|\alpha) = \frac{\Gamma(\sum_i \alpha_i)}{\prod_i \Gamma(\alpha_i)} \prod_i \theta_i^{\alpha_i} \Rightarrow \int_\theta \prod_i \theta_i^{\alpha_i} d\theta = \frac{\prod_i \Gamma(\alpha_i)}{\Gamma(\sum_i \alpha_i)} . \tag{9}$$

Model family 1: single multinomial. The model family H_1 models the contingency table $\mathbf{N}$ by a single multinomial distribution with parameters θ_{TY} that have a Dirichlet prior. Therefore

$$\begin{aligned} p(\mathbf{N}|H_1) &= \int_{\theta_{TY}} p(\mathbf{N}|H_1, \theta_{TY}) p(\theta_{TY}|H_1) d\theta_{TY} \\ &= \int_{\theta_{TY}} \left(\frac{N!}{\prod_{t,y} n_{ty}!} \prod_{t,y} \theta_{ty}^{n_{ty}} \right) \left(\frac{\Gamma(\sum_{ty} \alpha_{ty})}{\prod_{ty} \Gamma(\alpha_{ty})} \prod_{ty} \theta_{ty}^{\alpha_{ty}} \right) d\theta_{TY} \\ &= \frac{N!}{\prod_{t,y} n_{ty}!} \cdot \frac{\Gamma(\sum_{ty} \alpha_{ty})}{\prod_{ty} \Gamma(\alpha_{ty})} \cdot \int_{\theta_{TY}} \prod_{ty} \theta_{ty}^{n_{ty}+\alpha_{ty}} d\theta_{TY} \\ &= \frac{N!}{\prod_{t,y} n_{ty}!} \cdot \frac{\Gamma(\sum_{ty} \alpha_{ty})}{\prod_{ty} \Gamma(\alpha_{ty})} \cdot \frac{\prod_{ty} \Gamma(n_{ty} + \alpha_{ty})}{\Gamma(N + \sum_{ty} \alpha_{ty})} \end{aligned} \tag{10}$$

where the last line follows because the integral on the previous line is the integral of a Dirichlet distribution with parameters $n_{ty} + \alpha_{ty}$, without the normalizing factor.

Model family 2: independent margins. The model family H_2 models the contingency table by the product $\theta_T \theta_Y$ of independent multinomial distributions with parameters θ_T and θ_Y. Both θ_T and θ_Y have Dirichlet priors. Therefore

$$
\begin{aligned}
p(\mathbf{N}|H_2) &= \int_{\theta_T}\int_{\theta_Y} p(\mathbf{N}|H_2,\theta_T,\theta_Y)p(\theta_T|H_2)p(\theta_Y|H_2)d\theta_Y d\theta_T \\
&= \int_{\theta_T,\theta_Y}\left(\frac{N!}{\prod_{t,y} n_{ty}!}\prod_{t,y}(\theta_t\theta_y)^{n_{ty}}\right)\left(\frac{\Gamma(\sum_t \alpha_t)}{\prod_t \Gamma(\alpha_t)}\prod_t \theta_t^{\alpha_t}\right)\left(\frac{\Gamma(\sum_y \alpha_y)}{\prod_y \Gamma(\alpha_y)}\prod_t \theta_y^{\alpha_y}\right) d\theta_T d\theta_Y \\
&= \frac{N!}{\prod_{t,y} n_{ty}!}\cdot\frac{\Gamma(\sum_t \alpha_t)}{\prod_t \Gamma(\alpha_t)}\cdot\frac{\Gamma(\sum_y \alpha_y)}{\prod_y \Gamma(\alpha_y)}\cdot\int_{\theta_T}\prod_t \theta_t^{n_t+\alpha_t}d\theta_T\cdot\int_{\theta_Y}\prod_t \theta_y^{n_y+\alpha_y}d\theta_Y \\
&= \frac{N!}{\prod_{t,y} n_{ty}!}\cdot\frac{\Gamma(\sum_t \alpha_t)}{\prod_t \Gamma(\alpha_t)}\cdot\frac{\Gamma(\sum_y \alpha_y)}{\prod_y \Gamma(\alpha_y)}\cdot\frac{\prod_t \Gamma(n_t+\alpha_t)}{\Gamma(N+\sum_t \alpha_t)}\cdot\frac{\prod_y \Gamma(n_y+\alpha_y)}{\Gamma(N+\sum_y \alpha_y)}
\end{aligned} \tag{11}
$$

where the last line follows because the integrals on the previous line are of Dirichlet distributions without their normalizing factors.

Bayes factor between the families. The Bayes factor is the ratio of the marginal likelihoods, that is,

$$
\begin{aligned}
\frac{p(\mathbf{N}|H_1)}{p(\mathbf{N}|H_2)} &= \frac{N!}{\prod_{t,y} n_{ty}!}\cdot\frac{\Gamma(\sum_{ty}\alpha_{ty})}{\prod_{ty}\Gamma(\alpha_{ty})}\cdot\frac{\prod_{ty}\Gamma(n_{ty}+\alpha_{ty})}{\Gamma(N+\sum_{ty}\alpha_{ty})} \\
&\cdot\left(\frac{N!}{\prod_{t,y} n_{ty}!}\cdot\frac{\Gamma(\sum_t \alpha_t)}{\prod_t \Gamma(\alpha_t)}\cdot\frac{\Gamma(\sum_y \alpha_y)}{\prod_y \Gamma(\alpha_y)}\cdot\frac{\prod_t \Gamma(n_t+\alpha_t)}{\Gamma(N+\sum_t \alpha_t)}\cdot\frac{\prod_y \Gamma(n_y+\alpha_y)}{\Gamma(N+\sum_y \alpha_y)}\right)^{-1} \\
&= \frac{\prod_{ty}\Gamma(n_{ty}+\alpha_{ty})}{\prod_t \Gamma(n_t+\alpha_t)} \\
&\cdot\left(\frac{\Gamma(\sum_{ty}\alpha_{ty})}{\prod_{ty}\Gamma(\alpha_{ty})}\cdot\frac{\Gamma(N+\sum_t \alpha_t)}{\Gamma(N+\sum_{ty}\alpha_{ty})}\cdot\frac{\prod_t \Gamma(\alpha_t)}{\Gamma(\sum_t \alpha_t)}\cdot\frac{\prod_y \Gamma(\alpha_y)}{\Gamma(\sum_y \alpha_y)}\cdot\frac{\Gamma(N+\sum_y \alpha_y)}{\prod_y \Gamma(n_y+\alpha_y)}\right)
\end{aligned} \tag{12}
$$

where the terms on the last line are constant with respect to T. Dropping the constants yields (3).

B Interpretation of the Bayes factor

We start by computing a simple result for any two model families, and show that it can be applied to the Bayes factor in the fsIB cost function under a condition for the marginal priors.

A general result. Let H_1 and H_2 be any two families for generating a contingency table $\mathbf{N}$. Simple probability arithmetic yields the posterior probability of H_2 given H_1 and $\mathbf{N}$ as

$$
p(H_2|H_1,\mathbf{N}) = \frac{p(H_2,H_1,\mathbf{N})}{p(H_1,\mathbf{N})} = \frac{p(\mathbf{N}|H_1,H_2)}{p(\mathbf{N}|H_1)}\cdot p(H_2|H_1) \tag{13}
$$

where the first term on the right is (inverse to) the Bayes factor between H_1 and $H_1 \cap H_2$, and the second term is the prior probability of the family H_2 given H_1, which is independent of $\mathbf{N}$. Notice that the first term contains all the factors that depend on $\mathbf{N}$; it can therefore be called an *evidence* term.

Connection to fsIB. We wish to apply the above result to the Bayes factor in the fsIB cost function. The fsIB cost, however, compares families H_1 and H_2 directly, while the Bayes factor above compares H_1 and $H_1 \cap H_2$. A connection exists when $H_2 \subset H_1$, as analyzed below.

In the family H_2 the contingency table is generated from the product of some independent multinomial distributions θ_T and θ_Y for the marginals, whereas in H_1 it is generated from a multinomial joint distribution θ_{TY}. Since the product $\theta_T\theta_Y$ is also multinomial, it remains to ensure that $p(\theta_T\theta_Y|H_2) = p(\theta_T\theta_Y|H_1, H_2)$.

We have

$$p(\theta_T\theta_Y|H_1, H_2) = \frac{p(H_2|\theta_T\theta_Y, H_1)}{p(H_2|H_1)} p(\theta_T\theta_Y|H_1) = \frac{1}{p(H_2|H_1)} p(\theta_T\theta_Y|H_1) \qquad (14)$$

where the second equality follows since $\theta_T\theta_Y \in H_2$. When the prior $p(\theta_{TY}|H_1)$ is Dirichlet with parameters α_{ty}, this becomes

$$p(\theta_T\theta_Y|H_1, H_2) \propto \prod_{t,y} (\theta_t\theta_y)^{\alpha_{ty}} = \prod_t \theta_t^{\sum_y \alpha_{ty}} \prod_y \theta_y^{\sum_t \alpha_{ty}} \qquad (15)$$

which corresponds to separate Dirichlet priors for the marginals, with parameters $\alpha_t = \sum_y \alpha_{ty}$ and $\alpha_y = \sum_t \alpha_{ty}$. This yields the consistency condition in Section 2.1 for the marginal priors; note that the clustering does not affect the marginal of the words so only the condition for α_t matters. Given this condition, the result (13) states that the fsIB objective function is inverse to the evidence term in the posterior probability of independent margins.

A slight technical problem is that $p(H_2|H_1)$ is zero, but this can be corrected by introducing a suitable limit process.

C Connection between fsIB and mutual information

The objective function of fsIB is given in (3). We may equally optimize its logarithm. We may also add a constant term $\sum_y n_y \log n_y$ and divide by the total number of co-occurrences N; both are constant with respect to the clustering. This yields

$$\begin{aligned} &\frac{1}{N} \log C_{fsIB} + \frac{1}{N} \sum_y n_y \log n_y \\ &\qquad = \frac{1}{N} \sum_{t,y} \log\Gamma(n_{ty} + \alpha_{ty}) - \frac{1}{N} \sum_t \log\Gamma(n_t + \alpha_t) + \frac{1}{N} \sum_y n_y \log n_y \ . \qquad (16) \end{aligned}$$

By Stirling's approximation $\log\Gamma(x+1) = x\log(x) - x + O(\log(x))$, so the above can be rewritten as

$$\begin{aligned} &\frac{1}{N} \sum_{t,y} \left[(n_{ty} + \alpha'_{ty}) \log(n_{ty} + \alpha'_{ty}) - (n_{ty} + \alpha'_{ty}) + O(\log(n_{ty} + \alpha'_{ty})) \right] \\ &\qquad - \frac{1}{N} \sum_t \left[(n_t + \alpha'_t) \log(n_t + \alpha'_t) - (n_t + \alpha'_t) + O(\log(n_t + \alpha'_t)) \right] \\ &\qquad\qquad - \frac{1}{N} \sum_y n_y \log n_y \qquad (17) \end{aligned}$$

where $\alpha'_{ty} = \alpha_{ty} - 1$ and $\alpha'_t = \alpha_t - 1$.

The terms n_{ty}/N and n_t/N cancel each other. Applying the equality $(x+\epsilon)\log(x+\epsilon) = x\log(x) + x\log(1+\epsilon/x) + \epsilon\log(x+\epsilon)$, and noting that $n_t = \sum_y n_{ty}$ and $n_y = \sum_t n_{ty}$, the above can be further rewritten as

$$\begin{aligned}
&\sum_{t,y} \frac{n_{ty}}{N} \log \frac{n_{ty}}{n_t n_y} \\
&\qquad + \sum_{t,y} \left[\frac{\alpha'_{ty}}{N} \log(n_{ty} + \alpha'_{ty}) - \frac{\alpha'_{ty}}{N} + O(\frac{1}{N} \log(n_{ty} + \alpha'_{ty})) \right] \\
&\qquad\quad - \sum_{t} \left[\frac{\alpha'_t}{N} \log(n_t + \alpha'_t) - \frac{\alpha'_t}{N} + O(\frac{1}{N} \log(n_t + \alpha'_t)) \right] \\
&\qquad\qquad - \sum_{t,y} \left[\frac{n_{ty}}{N} \log(1 + \frac{\alpha'_{ty}}{n_{ty}}) - \frac{n_{ty}}{N} \log(1 + \frac{\alpha'_t}{n_t}) \right] \ . \qquad (18)
\end{aligned}$$

The first line is simply

$$\sum_{t,y} \hat{p}(t,y) \log \frac{\hat{p}(t,y)}{\hat{p}(t)\hat{p}(y)} - \log N = \hat{I}(T,Y) - \log N \ , \qquad (19)$$

that is, mutual information based on maximum likelihood probability estimates $\hat{p}(t,y) = n_{ty}/N$, $\hat{p}(t) = n_t/N$ and $\hat{p}(y) = n_y/N$, plus a constant term.

As N grows, $\hat{I}(T,Y)$ approaches the true mutual information $I(T,Y)$. By contrast, the terms on lines 2–4 of (18) go to zero. The terms on lines 2–3 go to zero because the denominator N grows faster than the logarithms and α_{ty} and α_t are constant. The terms on line 4 go to zero because n_{ty}/N and n_t/N are bounded and the terms inside the logarithms approach 1. Note that if n_{ty} or n_t were zero, the corresponding term on line 4 would be zero by definition.

Note on Stirling's approximation. The approximation used above applies for arguments above zero. Here we may have $n_{ty} + \alpha'_{ty} \le 0$ if $n_{ty} = 0$ for some y. However, then the corresponding term in (17) goes to zero as N grows and we may replace it by $\frac{1}{N} n_{ty} \log n_{ty} = 0$. This also leads to $\hat{I}(T,Y)$.

D Connection between fsIB and Jensen-Shannon divergence

When a document $\mathbf{d}$ with word co-occurrence counts d_y is added to cluster t, the ratio of the new and old fsIB objective function is given by (4). The cluster is chosen to maximize the ratio, or equivalently its logarithm, given by

$$\sum_y [\log\Gamma(m_{ty} + \alpha'_{ty} + 1) - \log\Gamma(n_{ty} + \alpha'_{ty} + 1)] - \log\Gamma(m_t + \alpha'_t + 1) + \log\Gamma(n_t + \alpha'_t + 1) \qquad (20)$$

where $\alpha'_{ty} = \alpha_{ty} - 1$, $\alpha'_t = \alpha_t$, $m_{ty} = n_{ty} + d_y$ and $m_t = n_t + |\mathbf{d}|$. We will now show that this equation is asymptotically equivalent to the cluster merging criterion (2) of the sequential IB algorithm, i.e., weighted Jensen-Shannon divergence between the word distributions in cluster and the document, weighted by their prior probabilities.

Assume initially that $n_{ty}+\alpha'_{ty} > 0$. Applying Stirling's approximation $\log\Gamma(x+1) = x\log(x) - x + O(\log(x))$, canceling the terms of type x and combining the terms of type $O(\log(x))$, the log ratio becomes

$$\begin{aligned}\sum_y [(m_{ty} + \alpha'_{ty})\log(m_{ty} + \alpha'_{ty}) - (n_{ty} + \alpha'_{ty})\log(n'_{ty} + \alpha'_{ty})]\\ - (m_t + \alpha'_t)\log(m_t + \alpha'_t) + (n_t + \alpha'_t)\log(n_t + \alpha'_t)\\ + \sum_y O(\log(m_{ty} + \alpha'_{ty})) + O(\log(m_t + \alpha'_t))\ . \end{aligned} \quad (21)$$

Applying the equality $(x+\epsilon)\log(x+\epsilon) = x\log(x) + x\log(1+\epsilon/x) + \epsilon\log(x+\epsilon)$, the above further becomes

$$\begin{aligned}&= \sum_y [m_{ty}\log m_{ty} - n_{ty}\log n_{ty}] - m_t\log m_t + n_t \log n_t\\ &+ \sum_y [m_{ty}\log(1+\frac{\alpha'_{ty}}{m_{ty}}) - n_{ty}\log(1+\frac{\alpha'_{ty}}{n_{ty}})] - m_t\log(1+\frac{\alpha'_t}{m_t}) + n_t\log(1+\frac{\alpha'_t}{n_t})\\ &+ \alpha'_{ty}\sum_y[\log(m_{ty}+\alpha'_{ty}) - \log(n_{ty}+\alpha'_{ty})] + \alpha'_t[\log(n_t+\alpha'_t) - \log(m_t+\alpha'_t)]\\ &+ \sum_y O(\log(m_{ty} + \alpha'_{ty})) + O(\log(m_t + \alpha'_t))\ .\end{aligned} \quad (22)$$

Let us add a term $-\sum_y d_y \log\frac{d_y}{|\mathbf{d}|}$, and divide by the total number of co-occurrences N; both are constant with respect to the cluster choice t. The above then becomes

$$\begin{aligned}&= \sum_y \left(\frac{m_{ty}}{N}\log\frac{m_{ty}}{m_t} - \frac{n_{ty}}{N}\log\frac{n_{ty}}{n_t} - \frac{d_y}{N}\log\frac{d_y}{|\mathbf{d}|}\right)\\ &+ \sum_y [\frac{m_{ty}}{N}\log(1+\frac{\alpha'_{ty}}{m_{ty}}) - \frac{n_{ty}}{N}\log(1+\frac{\alpha'_{ty}}{n_{ty}})] - \frac{m_t}{N}\log(1+\frac{\alpha'_t}{m_t}) + \frac{n_t}{N}\log(1+\frac{\alpha'_t}{n_t})\\ &+ \frac{\alpha'_{ty}}{N}\sum_y[\log(m_{ty}+\alpha'_{ty}) - \log(n_{ty}+\alpha'_{ty})] + \frac{\alpha'_t}{N}[\log(n_t+\alpha'_t) - \log(m_t+\alpha'_t)]\\ &+ \sum_y O(\frac{1}{N}\log(m_{ty} + \alpha'_{ty})) + O(\frac{1}{N}\log(m_t + \alpha'_t))\ .\end{aligned} \quad (23)$$

Let us first examine the first line closer. Since $m_{ty} = n_{ty} + d_y$, the first line can be rewritten as

$$\begin{aligned}&-\sum_y \left(\frac{n_{ty}}{N}\log\frac{n_{ty}/n_t}{m_{ty}/m_t} + \frac{d_y}{N}\log\frac{d_y/|\mathbf{d}|}{m_{ty}/m_t}\right)\\ &= -\hat{p}(t)\sum_y \hat{p}(y|t)\log\frac{\hat{p}(y|t)}{\bar{p}(y)} - \hat{p}(\mathbf{d})\sum_y \hat{p}(y|\mathbf{d})\log\frac{\hat{p}(y|\mathbf{d})}{\bar{p}(y)}\\ &= -(\hat{p}(t)+\hat{p}(\mathbf{d}))D_{JS}(\hat{p}(Y|t), \hat{p}(Y|\mathbf{d})|w_t, w_{\mathbf{d}})\end{aligned} \quad (24)$$

where we denoted

$$\begin{cases} \hat{p}(y|t) = n_{ty}/n_t \\ \hat{p}(y|\mathbf{d}) = d_y/|\mathbf{d}| \\ \hat{p}(t) = n_t/N \\ \hat{p}(\mathbf{d}) = |\mathbf{d}|/N \\ w_t = \hat{p}(t)/(\hat{p}(t)+\hat{p}(\mathbf{d})) \\ w_{\mathbf{d}} = \hat{p}(\mathbf{d})/(\hat{p}(t)+\hat{p}(\mathbf{d})) \\ \bar{p}(y) = m_{ty}/m_t = w_t\hat{p}(y|t) + w_{\mathbf{d}}\hat{p}(y|\mathbf{d}) \end{cases} . \tag{25}$$

Therefore, the first line of (23) is simply (opposite to) Jensen-Shannon divergence. The other terms arise due to the finite amount of data in the document and the cluster.

Let us examine what happens in the limit $N \to \infty$. The terms on the second line of (23) go to zero because $\frac{\alpha'_{ty}}{m_{ty}}$ and $\frac{\alpha'_t}{m_{ty}}$ go to zero and the multipliers before the logarithms are bounded above by 1. Note that if e.g. m_{ty} is zero, the corresponding term on the second line is zero by definition. The terms on the third and fourth lines also go to zero, since the denominator N grows faster than the logarithms. Note that the terms on lines 2–4 go to zero regardless of n_t and $|\mathbf{d}|$, i.e., even if the cluster and document grow along with N. Therefore, maximizing the fsIB cluster merging criterion (4) is asymptotically equivalent to minimizing the weighted Jensen-Shannon divergence (2).

Note on Stirling's approximation. In the above proof we applied Stirling's approximation to move from log-Gamma functions toward logarithms. Stirling's approximation for $\log\Gamma(x+1)$ applies when x is positive. We may assume that $m_{ty}+\alpha'_{ty}$, $m_t+\alpha'_t$ and $n_t+\alpha'_t$ are positive when N is large, but if $n_{ty}=0$ for some y we may have $-1 < n_{ty}+\alpha'_{ty} \le 0$.

However, if $n_{ty}=0$, the term $\frac{1}{N}\log\Gamma(n_{ty}+\alpha'_{ty}+1)$ goes to zero as N grows. To study the asymptotic behavior we may then simply replace it with $\frac{1}{N}n_{ty}\log n_{ty}=0$. This also leads to Jensen-Shannon divergence; the only difference is that the terms with n_{ty} are removed from lines 2–4 of (22) and (23) for that value of y.