Probabilisitic Graphical Models IFT6269 Project
Latent Dirichlet Allocation

Patrice Béchard [*1] **Théophile Gervet** [*2]

Abstract

TODO

1. Introduction and Related Works

1.1. Latent Dirichlet Allocation

1.1.1. Generative Process

Latent Dirichlet allocation (LDA) is a generative probabilisitic model of a corpus of D documents $\{\mathbf{w}_1, ..., \mathbf{w}_D\}$. Each document $\mathbf{w}_d = (w_{d,1}, ..., w_{d,N_d})$ is a sequence of N_d words from a vocabulary indexed by $\{1, ..., V\}$.

The basic idea behind LDA is that each document is represented as a random mixture over K latent topics, where each topic is characterized by a multinomial distribution over words, parametrized by a vector $\boldsymbol{\beta}_k$, $k \in 1, ..., K$.

Formally, LDA assumes the following generative process for each document $\mathbf{w}_d$:

1. Sample a distribution over topics $\boldsymbol{\theta_d} \sim Dir(\boldsymbol{\alpha})$
2. For each of the N_d words $w_{d,n}$ independently:
   (a) Draw a topic from the distribution over topics $z_{d,n} \sim \text{Mult}(\boldsymbol{\theta}_d)$
   (b) Draw a word from this topic $w_{d,n} \sim \text{Mult}(\boldsymbol{\beta}_{z_{d,n}})$

This generative process is illustrated in the directed graphical model of figure 1.

The joint distribution associated with this model is:

$$\prod_{k=1}^{K} p(\boldsymbol{\beta}_k|\boldsymbol{\eta}) \prod_{d=1}^{D} p(\boldsymbol{\theta}_d|\boldsymbol{\alpha}) \prod_{n=1}^{N_d} p(z_{d,n}|\boldsymbol{\theta}_d) p(w_{d,n}|z_{d,n}, \boldsymbol{\beta}_{1:K}) \tag{1}$$

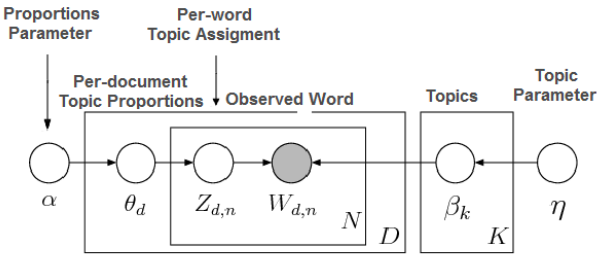

Figure 1. Graphical model corresponding to the generative process for LDA.

where
$z_{d,n} \sim \text{Mult}(\boldsymbol{\theta}_d)$, i.e $p(z_{d,n}|\boldsymbol{\theta}_d) = \theta_{d,z_{d,n}}$
$w_{d,n} \sim \text{Mult}(\boldsymbol{\beta}_{z_{d,n}})$, i.e $p(w_{d,n}|z_{d,n}, \boldsymbol{\beta}_{1:K}) = \beta_{z_{d,n},w_{d,n}}$
$\boldsymbol{\beta}_k \sim \text{Dir}(\boldsymbol{\eta})$
$\boldsymbol{\theta}_d \sim \text{Dir}(\boldsymbol{\alpha})$

This generative process is just a model of the structure we assume to be in our data. In reality, we only observe documents $\mathbf{w}_{1:D}$ and our goal is to infer the underlying topic structure: the topics $\boldsymbol{\beta}_{1:K}$, the per document distributions over topics $\boldsymbol{\theta}_d$, and per document per word topic assignments $z_{d,n}$.

Since the generative process assumes independence between documents, the probability of the whole corpus decomposes as a product of terms for individual documents. In order to clarify the notation, from now on let us consider equations for a single document $\mathbf{w}$ with N words.

Assuming the Dirichlet hyper-parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\eta}$ are fixed, we want to infer the posterior distribution:

$$p(\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{z}|\mathbf{w}, \boldsymbol{\alpha}, \boldsymbol{\eta}) = \frac{p(\boldsymbol{\beta}, \boldsymbol{\theta}, \mathbf{z}, \mathbf{w}|\boldsymbol{\alpha}, \boldsymbol{\eta})}{p(\mathbf{w}|\boldsymbol{\alpha}, \boldsymbol{\eta})} \tag{2}$$

Unfortunately, this distribution is intractable to compute because the normalization factor $p(\mathbf{w}|\boldsymbol{\alpha}, \boldsymbol{\eta})$ cannot be computed exactly. We must use approximate inference to estimate the posterior over latent variables $\boldsymbol{\beta}$, $\boldsymbol{\theta}$ and $\mathbf{z}$. We will explore both variational inference and Gibbs sampling

[*]Equal contribution [1]Université de Montréal, Montréal, Canada [2]McGill University, Montréal, Canada. Correspondence to: Patrice Béchard <patrice.bechard@umontreal.ca>, Théophile Gervet <theophile.gervet@umontreal.ca>.

Proceedings of the 34^{th} International Conference on Machine Learning, Sydney, Australia, PMLR 70, 2017.

to solve this problem.

1.2. Related Works

Most of the work done in this article is based on the original Latent Dirichlet Allication paper (**?**), which covers the general theory of the model as well as the variational inference approach to estimate the posterior. Other works present the Gibbs sampling approach for the same problem and provide useful implementation details (**???**).

Latent Dirichlet Allocation is an extension of the probabilistic latent semantic analysis (pLSA) model (**?**), which postulates that a document $\mathbf{w}$ and a word w_n are conditionally independent given a latent topic z :

$$p(\mathbf{w}, w_n) = p(\mathbf{w}) \sum_z p(w_n|z)p(z|\mathbf{w})$$

Some problems arise with this model. First, while it does make it possible for a document to contain multiple topics, the model only learns the topic mixtures $p(z|\mathbf{w})$ for the documents it has seen during the training, making it difficult to use it to assign probability to previously unseen documents. Furthermore, the number of parameters which the model tries to estimate grows linearly with the number of documents in the training set, which makes the model prone to overfitting. LDA overcomes both of the main problems with pLSA by adding a Dirichlet prior over the documents, defining a complete generative model.

However, LDA is by no means perfect. There have been many new ways to enhance the model to make it more realistic by relaxing some assumptions made by the model. The bag-of-words assumption is fine for our use of LDA, which is to uncover the topics present in a document and to assign a certain word to a topic. For more complicated goals, such as generating text, the assumption becomes inappropriate. There have been a number of extensions of the model that relax the bag-of-words assumption, for example, by combining the LDA with a Hidden Markov Model (HMM) to take into account the ordering of the words (**?**). It is also possible to relax the assumption that the ordering of the documents is irrelevant if we wish to find the evolution of certain topics over time, for example. A dynamic topic model would then be more appropriate than the classic LDA (**?**).

2. Algorithms

2.1. Variational EM

To make our life easier, we derive a variational EM algorithm for a slightly simpler graphical model for LDA (the one presented in the original paper). We remove the Dirichlet prior over topics, this corresponds to removing the dependence of the $\boldsymbol{\beta}$ plate on the $\boldsymbol{\eta}$ random variable (see Figure 2, left). By doing so, we lose our ability to smooth topics through the $\boldsymbol{\eta}$ parameter.

Assuming the $\boldsymbol{\alpha}$ parameter and topics $\boldsymbol{\beta}$ are fixed, the posterior for a single document $\mathbf{w}$ now takes the form:

$$p(\boldsymbol{\theta}, \mathbf{z}|\mathbf{w}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \frac{p(\boldsymbol{\theta}, \mathbf{z}, \mathbf{w}|\boldsymbol{\alpha}, \boldsymbol{\beta})}{p(\mathbf{w}|\boldsymbol{\alpha}, \boldsymbol{\beta})} \tag{3}$$

This expression is still intractable because of the denominator $p(\mathbf{w}|\boldsymbol{\alpha}, \boldsymbol{\beta})$.

Let us motivate a variational EM algorithm by looking at how this intractable posterior comes back to bite us when we try to maximize the likelihood of the observed data. Since the probability of the whole corpus decomposes as a product of terms for individual documents, the log likelihood of the whole corpus decomposes as a sum of terms for individual documents. The log likelihood of a single document $\mathbf{w}$ takes the form:

$$\log p(\mathbf{w}|\boldsymbol{\alpha}, \boldsymbol{\beta}) = \log \int \sum_{\mathbf{z}} p(\boldsymbol{\theta}, \mathbf{z}, \mathbf{w}|\boldsymbol{\alpha}, \boldsymbol{\beta}) \mathrm{d}\boldsymbol{\theta} \tag{4}$$

Because of the integral over $\boldsymbol{\theta}$ and the sum over $\mathbf{z}$ inside the log, we cannot maximize this objective directly. One common way to solve this problem is the Expectation Maximization (EM) algorithm. We make the following observation:

$$\begin{aligned}
\log p(\mathbf{w}|\boldsymbol{\alpha}, \boldsymbol{\beta}) &= \log \int \sum_{\mathbf{z}} \frac{p(\boldsymbol{\theta}, \mathbf{z}, \mathbf{w}|\boldsymbol{\alpha}, \boldsymbol{\beta}) q(\boldsymbol{\theta}, \mathbf{z})}{q(\boldsymbol{\theta}, \mathbf{z})} \mathrm{d}\boldsymbol{\theta} \\
&= \mathbb{E}_q \left[\log \int \sum_{\mathbf{z}} \frac{p(\boldsymbol{\theta}, \mathbf{z}, \mathbf{w}|\boldsymbol{\alpha}, \boldsymbol{\beta})}{q(\boldsymbol{\theta}, \mathbf{z})} \mathrm{d}\boldsymbol{\theta} \right] \\
&\geq \mathbb{E}_q \left[\log p(\boldsymbol{\theta}, \mathbf{z}, \mathbf{w}|\boldsymbol{\alpha}, \boldsymbol{\beta}) \right] - \mathbb{E}_q \left[\log q(\boldsymbol{\theta}, \mathbf{z}) \right] \\
&= \mathcal{L}(q, \boldsymbol{\alpha}, \boldsymbol{\beta})
\end{aligned}$$

where we have introduced an arbitrary distribution $q(\boldsymbol{\theta}, \mathbf{z})$ over the problematic latent variables $\boldsymbol{\theta}$ and $\mathbf{z}$, and the last step follows from Jensen's inequality.

The EM algorithm consists in alternatively maximizing the lower bound $\mathcal{L}$ on the log likelihood with respect to the distribution q (the E-step), and with respect to parameters $\boldsymbol{\alpha}$ and $\boldsymbol{\beta}$ (the M-step).

One can easily show that maximizing $\mathcal{L}$ with respect to q is equivalent to minimizing $\mathrm{KL}\left(q(\boldsymbol{\theta}, \mathbf{z}) || p(\boldsymbol{\theta}, \mathbf{z}|\mathbf{w}, \boldsymbol{\alpha}, \boldsymbol{\beta})\right)$. A simple solution is thus to set $q(\boldsymbol{\theta}, \mathbf{z})$ to be $p(\boldsymbol{\theta}, \mathbf{z}|\mathbf{w}, \boldsymbol{\alpha}, \boldsymbol{\beta})$, the true posterior over latent variables under the model.

In our case, things are not so simple: as we have seen, the posterior $p(\boldsymbol{\theta}, \mathbf{z}|\mathbf{w}, \boldsymbol{\alpha}, \boldsymbol{\beta})$ is intractable. This is were variational inference comes into play. The basic idea is to introduce a parametrized family of distributions over latent variables and phrase inference as an optimization problem.

We parametrize q as follows, according to the mean field independence assumption:

$$q(\boldsymbol{\theta}, \mathbf{z}|\boldsymbol{\gamma}, \boldsymbol{\phi}) = q(\boldsymbol{\theta}|\boldsymbol{\gamma}) \prod_{n=1}^{N} q(z_n|\boldsymbol{\phi}_n) \tag{5}$$

where the dirichlet parameters $\boldsymbol{\gamma}$ and the multinomial parameters $\boldsymbol{\phi}_n$, $n \in \{1, ..., N\}$ are free variational parameters. The graphical model corresponding to this parameterization is illustrated in Figure 2, right.

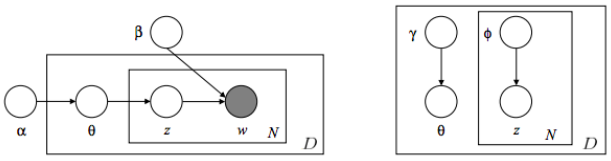

Figure 2. (Left) Graphical model representation of LDA without prior over topics. (Right) Graphical model representation of the variational distribution used to approximate the posterior.

Now variational inference consists in solving the following optimization problem:

$$(\boldsymbol{\gamma}^*, \boldsymbol{\phi}^*) = \arg\min_{\boldsymbol{\gamma}, \boldsymbol{\phi}} \mathrm{KL}(q(\boldsymbol{\theta}, \mathbf{z}|\boldsymbol{\gamma}, \boldsymbol{\phi}) \,||\, p(\boldsymbol{\theta}, \mathbf{z}|\mathbf{w}, \boldsymbol{\alpha}, \boldsymbol{\beta})) \tag{6}$$

Note that $q(\boldsymbol{\theta}, \mathbf{z}|\mathbf{w}, \boldsymbol{\gamma}^*, \boldsymbol{\phi}^*)$ is actually a conditional distribution varying as a function of $\mathbf{w}$ because the optimization objective depends on $\mathbf{w}$. It can thus be seen as an approximation to the posterior $p(\boldsymbol{\theta}, \mathbf{z}|\mathbf{w}, \boldsymbol{\alpha}, \boldsymbol{\beta})$.

With this tool at our disposal, we can come back and complete the EM algorithm described in section 1.1.2. We can expand our variational lower bound on the per-document log likelihood as follows:

$$\mathcal{L}(\boldsymbol{\gamma}, \boldsymbol{\phi}, \boldsymbol{\alpha}, \boldsymbol{\beta}) = \mathbb{E}_q[\log p(\boldsymbol{\theta}|\boldsymbol{\alpha})] + \mathbb{E}_q[\log p(\mathbf{z}|\boldsymbol{\theta})] + \mathbb{E}_q[\log p(\mathbf{w}|\mathbf{z}, \boldsymbol{\beta})] - \mathbb{E}_q[\log q(\boldsymbol{\theta}|\boldsymbol{\gamma})] - \mathbb{E}_q[\log q(\mathbf{z}|\boldsymbol{\phi})] \tag{7}$$

where the first three terms correspond to the expected complete log likelihood under q and the last two terms to the entropy of q.

2.1.1. E-STEP

In the E-step, we maximize $\mathcal{L}$ with respect to the variational parameters $\boldsymbol{\gamma}$ and $\boldsymbol{\phi}$. This corresponds to applying the following update rules iteratively until convergence for each document:

$$\boldsymbol{\phi}_{n,k} \propto \boldsymbol{\beta}_{k,w_n} \exp\{\Psi(\boldsymbol{\gamma}_k)\} \tag{8}$$

$$\boldsymbol{\gamma}_k = \boldsymbol{\alpha}_k + \sum_{n=1}^{N} \boldsymbol{\phi}_{n,k} \tag{9}$$

where $\Psi(x) = \frac{\Gamma'(x)}{\Gamma(x)}$ is the digamma function.

Theses updates make sense intuitively. Equation 8 tells us that $\boldsymbol{\phi}_{n,k}$, the probability that word w_n is generated by latent topic k, is proportional to the probability of drawing w_n from topic k times a quantity proportional to how much the document likes topic k (the dirichlet parameter $\boldsymbol{\gamma}_k$). Equation 9 tells us that the dirichlet parameter $\boldsymbol{\gamma}_k$ is equal to the pseudo-count $\boldsymbol{\alpha}_k$ given by the dirichlet prior plus the expected count of words picking topic k.

2.1.2. M-STEP

For the M-step, we maximize the overall variational lower bound with respect to parameter $\boldsymbol{\beta}$. We could also maximize with respect to $\boldsymbol{\alpha}$ but we chose to keep it fixed to make things simpler. As said before, the overall log likelihood of the corpus is the sum of the log likelihoods for the individual documents. Likewise, the overall variational lower bound is the sum of the individual variational bounds.

Maximizing with respect to $\boldsymbol{\beta}$ gives the following update:

$$\boldsymbol{\beta}_{k,j} \propto \sum_{d=1}^{D} \sum_{n=1}^{N_d} \boldsymbol{\phi}_{d,n,k} w_{d,n}^{(j)} \tag{10}$$

where we abuse notation and index word $w_{d,n}$ which was a scalar until here as a one hot vector.

Equation 10 also has an intuitive meaning: the probability of word j under topic k is proportional to the expected count under the variational distribution of the number of times word j is assigned topic k across the whole corpus.

2.1.3. OVERALL ALGORITHM

We implemented a variational "EM flavored" algorithm which we described in detail in algorithm 1. Note that to be strictly an EM algorithm, we should maximize $\mathcal{L}$ with respect to $\boldsymbol{\phi}$ and $\boldsymbol{\gamma}$ all the way at each E-step. Instead, we alternate a single step of improvement with respect to the variational parameters with one step of improvement with respect to $\boldsymbol{\beta}$. Our algorithm has the same guarantee to increase the variational lower bound at each iteration, it is just more convenient to check for convergence only for the overall algorithm instead of for each E-step.

2.2. Gibbs Sampling

Once again, we consider a single document $\mathbf{w}$ with N words. Recall that we are interested in the latent topics $\boldsymbol{\beta}_{1:K}$, the distribution over topics $\boldsymbol{\theta}$, and the per word topic assignments z_n, $n \in \{1, ..., N\}$.

Algorithm 1 Variational EM

Input: documents $\mathbf{w}_1, ..., \mathbf{w}_D$
Set $\boldsymbol{\alpha}_k = 1/K$ for all k
Randomly initialize variational parameters $\boldsymbol{\gamma}$, $\boldsymbol{\phi}$
repeat
  for $d = 1$ **to** D **do**
    for $k = 1$ **to** K **do**
      for $n = 1$ **to** N_d **do**
        $\boldsymbol{\phi}_{d,n,k} = \boldsymbol{\beta}_{k,w_n} \exp\{\Psi(\boldsymbol{\gamma}_{d,k})\}$
      end for
      normalize $\boldsymbol{\phi}_{d,n}$ to sum to 1
      $\boldsymbol{\gamma}_{d,k} = \boldsymbol{\alpha}_k + \sum_{n=1}^{N_d} \boldsymbol{\phi}_{d,n,k}$
    end for
  end for
  for $k = 1$ **to** K **do**
    for $j = 1$ **to** V **do**
      $\boldsymbol{\beta}_{k,j} = \sum_{d=1}^{D} \sum_{n=1}^{N_d} \boldsymbol{\phi}_{d,n,k} w_{d,n}^{(j)}$
    end for
    normalize $\boldsymbol{\beta}_k$ to sum to 1
  end for

until objective $\mathcal{L}$ doesn't improve anymore

We could derive conditional distributions for each of these latent variables, and therefore an LDA Gibbs sampling algorithm, but we note that both $\boldsymbol{\theta}_d$ and $\boldsymbol{\beta}_k$ can be computed using the topic assignments $z_{d,n}$ which are sufficient statistics. This allows us to integrate out the multinomial parameters and simply sample $z_{d,n}$. This strategy is referred to as *collapsed* or *Rao-Blackwellized* Gibbs sampling (**?**).

The conditional we want to sample the topic assignment z_n from is given by :

$$p(z_n|\mathbf{z}_{\neg n}, \boldsymbol{\alpha}, \boldsymbol{\eta}, \mathbf{w})$$

where $\mathbf{z}_{\neg n}$ denotes all topic allocations except for z_n and where we have reintroduced the hyperparameter $\boldsymbol{\alpha}$ controlling the topics-per-document proportions as well as the hyperparameter $\boldsymbol{\eta}$ controlling the probability of a word being assign to a topic. We can develop this expression using Bayes' rule:

$$p(z_n|\mathbf{z}_{\neg n}, \mathbf{w}, \boldsymbol{\alpha}, \boldsymbol{\eta}) \propto p(w_n|z_n, \mathbf{z}_{\neg n}, \mathbf{w}_{\neg n}, \boldsymbol{\eta}) p(z_n|\mathbf{z}_{\neg n}, \boldsymbol{\alpha}) \tag{11}$$

where we have used the fact that $\mathbf{w} \perp \boldsymbol{\alpha} | \mathbf{z}$ and $\mathbf{z} \perp \boldsymbol{\eta}$. Now, let's develop each term of equation 11 for a given topic assignment $z_n = k$:

$$\begin{aligned} p(w_n|z_n = k, \mathbf{z}_{\neg n}, \mathbf{w}_{\neg n}) &= \int p(w_n|z_n = k, \boldsymbol{\beta}_k) p(\boldsymbol{\beta}_k|\mathbf{z}_{\neg n}, \mathbf{w}_{\neg n}) \mathrm{d}\boldsymbol{\beta}_k \\ &= \frac{n_{\neg n,k}^{(w_n)} + \boldsymbol{\eta}_n}{\sum_{n'=1}^{V} \left(n_{\neg n,k}^{(w_{n'})} + \boldsymbol{\eta}_{n'} \right)} \end{aligned} \tag{12}$$

$$\begin{aligned} p(z_n = k|\mathbf{z}_{\neg n}) &= \int p(z_n = k|\theta_{\mathbf{w}_n}) p(\theta_{\mathbf{w}_n}|\mathbf{z}_{\neg n}) \mathrm{d}\theta_{\mathbf{w}_n} \\ &= \frac{n_{\neg n,k}^{(\mathbf{w}_n)} + \boldsymbol{\alpha}_k}{\sum_{k'=1}^{(K)} \left(n_{\neg n,k'}^{(\mathbf{w}_n)} + \boldsymbol{\alpha}_{k'} \right)} \end{aligned} \tag{13}$$

The results obtained follow a Dirichlet $\left(n_{\neg n,k}^{(w_n)} + \boldsymbol{\eta}_n \right)$ and a Dirichlet $\left(n_{\neg n,k}^{(\mathbf{w}_n)} + \boldsymbol{\alpha}_k \right)$, respectively. These distributions were obtained by the integration of the product between a Multinomial distribution and its Dirichlet conjugate prior. A detailed derivation of these equations is presented in (**?**).

Putting together equations 12 and 13, we obtain the posterior distribution from which we sample a new topic assignment z_n :

$$\begin{aligned} p(z_n = k|\mathbf{z}_{\neg n}, \mathbf{w}) &\propto \frac{n_{\neg n,k}^{(w_n)} + \boldsymbol{\eta}_n}{\sum_{n'=1}^{V} \left(n_{\neg n,k}^{(w_{n'})} + \boldsymbol{\eta}_{n'} \right)} \frac{n_{\neg n,k}^{(\mathbf{w}_n)} + \boldsymbol{\alpha}_k}{\sum_{k'=1}^{(K)} \left(n_{\neg n,k'}^{(\mathbf{w}_n)} + \boldsymbol{\alpha}_{k'} \right)} \end{aligned} \tag{14}$$

Finally, after the *burn-in* period of the Gibbs sampling, we can find the expressions for the parameters we are looking for, namely the term-by-topic distribution $\boldsymbol{\beta}$ and the topic-by-document distribution $\boldsymbol{\theta}$ as such:

$$\boldsymbol{\beta}_{k,n} = \frac{n_k^{(w_n)} + \boldsymbol{\eta}_t}{\sum_{t=1}^{V} n_k^{(t)} + \boldsymbol{\eta}_t} \tag{15}$$

$$\boldsymbol{\theta}_{d,k} = \frac{n_m^{(k)} + \boldsymbol{\alpha}_k}{\sum_{k=1}^{K} n_m^{(k)} + \boldsymbol{\alpha}_k} \tag{16}$$

Algorithm 2 presents the various steps of the implementation of LDA using collapsed Gibbs sampling to estimate the posterior distribution.

Algorithm 2 Gibbs Sampling

Input: documents $\mathbf{w}_1, ..., \mathbf{w}_D$
Initialize hyperparameters $\boldsymbol{\alpha}$ and $\boldsymbol{\eta}$
Randomly initialize topic assignments $z_{d,n}$ for all d, n
Increment counters $n_{d,k}$ and $n_{k,n}$
repeat
  for $d = 1$ **to** D **do**
    for $n = 1$ **to** N_d **do**
      $n_{d,z_{d,n}} -= 1; \quad n_{z_{d,n},n} -= 1$
      Sample topic assignment $z_{d,n}$ from equation 14
      $n_{d,z_{d,n}} += 1; \quad n_{z_{d,n},n} += 1$
    end for
  end for
until convergence
compute $\boldsymbol{\beta}_{k,n}$ from equation 15 for all d, k
compute $\boldsymbol{\theta}_{d,k}$ from equation 16 for all k, n

2.2.1. Choice of hyperparameters

Both $\boldsymbol{\alpha}$ and $\boldsymbol{\eta}$ are hyperparameters for our model. Typically, the a priori assumption of LDA is that all words are equally likely to be assigned to a topic, and all topics have equal chance of being assigned to a document. In this case, values of $50/K$ for each component of $\boldsymbol{\alpha}$ and 0.01 for each component of $\boldsymbol{\eta}$ have been reported to yield good results (**?**).

3. Experimental Methods

For experiments with both implementations of LDA, we used the *20 Newsgroups* dataset (**?**). We train the models on a subset of the dataset, and assess generalization to unseen data using the *perplexity* metric. In order to infer the distribution over topics for documents unseen during training (the $\boldsymbol{\theta}$ parameter) with Gibbs sampling, we use a method called query sampling. With the model trained with variational EM, we simply use variational inference.

3.1. Dataset and Preprocessing

The *20 Newsgroups* dataset contains approximately 20,000 newsgroup documents, partitioned nearly evenly across 20 different newsgroups, each corresponding to a different subject. Some of the newsgroups are very closely related to each other while others are unrelated. Figure 3 roughly partitions the 20 newsgroups by subject matter.

We use the train-test partition provided by sklearn (60%-40%), and preprocess the data by tokenizing, removing stop-words and punctuation, lemmatizing with the WordNetLemmatizer from the nltk library, and keeping only the 3000 most frequent words.

comp.graphics comp.os.ms-windows.misc comp.sys.ibm.pc.hardware comp.sys.mac.hardware comp.windows.x	rec.autos rec.motorcycles rec.sport.baseball rec.sport.hockey	sci.crypt sci.electronics sci.med sci.space
misc.forsale	talk.politics.misc talk.politics.guns talk.politics.mideast	talk.religion.misc alt.atheism soc.religion.christian

Figure 3. Newsgroups classified by overlap in subjects

3.2. Perplexity

Perplexity is a measure of the generalization performance of models and is used by convention in language modeling. It is defined algebraically as the inverse of the geometric mean per-word likelihood. A lower perplexity score indicates a better generalization performance for a model. For a test set of D documents, the perplexity is given by :

$$\begin{aligned}\text{perplexity}(D_{\text{train}}) &= \sum_{d=1}^{D} p(\mathbf{w_d})^{-1/N_d} \\ &= \exp\left[-\frac{\sum_{d=1}^{D}\log p(\mathbf{w}_d)}{\sum_{d=1}^{D} N_d}\right] \end{aligned} \tag{17}$$

For LDA, the likelihood found $p(\mathbf{w}_d)$ found in equation 17 can be expressed as the parameters $\boldsymbol{\beta}$ and $\boldsymbol{\theta}$. If we develop the expression for the likelihood, we have :

$$\begin{aligned} p(\mathbf{w}_d) &= \prod_{n=1}^{N_d}\sum_{k=1}^{K} p(w_{d,n} = t|z_{n,d} = k)\cdot p(z_{d,n} = k|d = d) \\ &= \prod_{t=1}^{V}\left(\sum_{k=1}^{K}\boldsymbol{\beta}_{k,t}\cdot\boldsymbol{\theta}_{d,k}\right)^{n_d^{(t)}} \\ \log p(\mathbf{w}_d) &= \sum_{t=1}^{V} n_d^{(t)}\log\left(\sum_{k=1}^{K}\boldsymbol{\beta}_{k,t}\cdot\boldsymbol{\theta}_{d,k}\right) \end{aligned} \tag{18}$$

Here, $n_d^{(t)}$ is the number of times term t appears in document d. Note here that the term $\boldsymbol{\theta}_{d,k}$ has to be obtained for the new test documents via methods shown in the subsequent sections.

3.3. Query sampling

In order to infer the distribution over topics for documents unseen during training (the $\boldsymbol{\theta}$ parameter) with Gibbs sampling, we use a method called query sampling.

We consider the query (vector of words) $\tilde{\mathbf{w}}$ (note that we can have multiple queries at the same time, but we only consider one here for simplicity). Normally, the task of querying consists of finding similarity between the query and other documents by estimating the posterior distribution $\tilde{\mathbf{z}}$ given the query and the state of the LDA at the end of the training (represented by the Markov state $\boldsymbol{\mathcal{M}} = \{\mathbf{z}, \mathbf{w}\}$). Here, we want to find this distribution to compute the perplexity of our model. The algorithm is pretty straightforward and consists of assign topics randomly to words in $\tilde{\mathbf{w}}$ and to perform a certain number of loops through Gibbs sampling update on our query only. The distribution from which we sample $\tilde{z}$ from is given below :

$$p(\tilde{z}_n = k | \tilde{w}_n = t, \tilde{\mathbf{z}}_{\neg n}, \tilde{\mathbf{w}}_{\neg n}; \boldsymbol{\mathcal{M}}) = \frac{n_k^{(t)} + \tilde{n}_{k,\neg n}^{(t)} + \boldsymbol{\eta}_t}{\sum_{t'=1}^{V} n_k^{(t')} + \tilde{n}_{k,\neg n}^{(t')} + \boldsymbol{\eta}'_t} \cdot \frac{n_{\tilde{d},\neg n}^{(k)} + \boldsymbol{\alpha}_k}{\sum_{k'=1}^{K} n_{\tilde{d}(k')} + \boldsymbol{\alpha}'_k} \tag{19}$$

where $\tilde{n}_k^{(t)}$ is the number of observations of term t and topic k in the query. With this posterior distribution, we obtain the topic-per-document distribution for the document:

$$\boldsymbol{\theta}_{\tilde{m},k} = \frac{n_{\tilde{m}}^{(k)} + \boldsymbol{\alpha}_k}{\sum_{k'=1}^{K} n_{\tilde{m}}^{(k')} + \boldsymbol{\alpha}'_k} \tag{20}$$

It is important that the size of the unknown documents be smaller than the training set, because the posterior (equation 19) distorts the topic distribution obtained from the training.

4. Results

4.1. Choosing the Number of Topics

We pick the number of topics which gives the best generalization performance (the best test-set perplexity).

For the variational inference implementation, we ran into some numerical issues to compute the likelihood of documents. Thus we computed training and test set perplexity with the batch variational inference implementation by sklearn. As we can see in figure 4, training-set perplexity always goes down when we increase the number of topics, but the best test-set perplexity is attained for 12 topics (for more topics, we overfit).

TODO add similar curve for implementation with Gibbs sampling

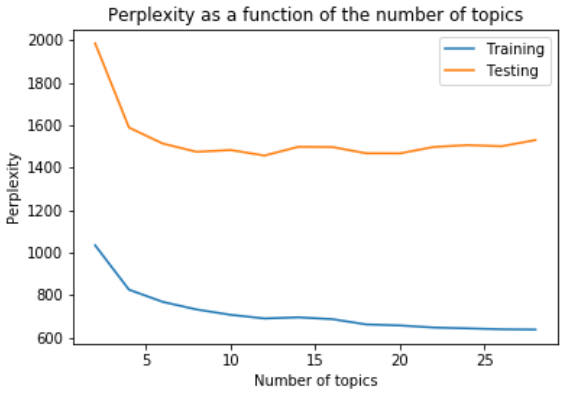

Figure 4. Variational inference implementation: the best test set perplexity is reached at 12 topics.

4.2. Visual Inspection of Topics

We trained both our models with 12 topics in order to visually inspect the learnt topics. The most probable words for each topic for LDA trained with our model of variational EM are in table 1.

Table 1. Most probable words associated to some topics for LDA trained with variational EM.

SPACE	COMPUTER	RELIGION	NUMBERS	CYBER SECURITY	GEN
space	use	god	10	key	arm
nasa	windows	people	00	government	pe
program	file	believe	25	encryption	j
earth	card	jesus	15	privacy	tu
launch	window	know	11	security	

TODO put table at the top of the page on both columns (don't know how to do)
TODO add similar table for implementation with Gibbs sampling