Fast Inference and Learning for Modeling Documents with a Deep Boltzmann Machine

Nitish Srivastava NITISH@CS.TORONTO.EDU
Ruslan Salakhutdinov RSALAKHU@CS.TORONTO.EDU
Geoffrey Hinton HINTON@CS.TORONTO.EDU
University of Toronto, 6 Kings College Road, Toronto, ON M5S 3G4 CANADA

Abstract

We introduce a type of Deep Boltzmann Machine (DBM) that is suitable for extracting distributed semantic representations from a large unstructured collection of documents. We propose an approximate inference method that interacts with learning in a way that makes it possible to train the DBM more efficiently than previously proposed methods. Even though the model has two hidden layers, it can be trained just as efficiently as a standard Restricted Boltzmann Machine. Our experiments show that the model assigns better log probability to unseen data than the Replicated Softmax model. Features extracted from our model outperform LDA, Replicated Softmax, and DocNADE models on document retrieval and document classification tasks.

1. Introduction

Text documents are a ubiquitous source of information. Representing the information content of a document in a form that is suitable for solving real-world problems is an important task. The aim of topic modeling is to create such representations by discovering latent topic structure in collections of documents. These representations are useful for document classification and retrieval tasks, making topic modeling an important machine learning problem.

The most common approach to topic modeling is to build a generative probabilistic model of the bag of words in a document. Directed graphical models such as Latent Dirichlet Allocation (LDA), CTM, H-LDA have been extensively used for this (Blei et al., 2003; 2010; Mimno & McCallum, 2008). Non-parametric extensions of these models have also been quite successful (Teh et al., 2006; Blei, 2012; Griffiths & Steyvers, 2004). Even though exact inference in these models is hard, efficient inference schemes, including stochastic variational inference, online inference, and collapsed Gibbs have been developed that make it feasible to train and use these methods (Teh et al., 2008; Wang & Blei, 2009; Canini et al., 2009). Another approach is to use undirected graphical models such as the Replicated Softmax model (Salakhutdinov & Hinton, 2009b). In this model, inferring latent topic representations is exact and efficient. However, training is still hard and often requires careful hyperparameter selection. These models typically perform better than LDA in terms of both the log probability they assign to unseen data and their document retrieval and document classification accuracy. Recently, neural network based approaches such as Neural Autoregressive Density Estimators (DocNADE) (Larochelle & Lauly, 2012) have been to shown to outperform the Replicated Softmax model.

The Replicated Softmax model is a family of Restricted Boltzmann Machines (RBMs) with shared parameters. An important feature of RBMs is that they solve the "explaining-way" problem of directed graphical models by having a complementary prior over hidden units. However, this implicit prior may not be the best prior to use and having some degree of flexibility in defining the prior may be advantageous. One way of adding this additional degree of flexibility, while still avoiding the explaining-away problem, is to learn a two hidden layer Deep Boltzmann Machine (DBM). This model adds another layer of hidden units on top of the first hidden layer with bi-partite, undirected connections. The new connections come with a new set of weights. However, this additional implicit prior comes

Presented at the International Conference on Machine Learning (ICML) workshop on *Inferning: Interactions between Inference and Learning*, Atlanta, Georgia, USA, 2013.

at the cost of more expensive training and inference. Therefore, we have the following two extremes: On one hand, RBMs can be efficiently trained (e.g. using Contrastive Divergence), inferring the state of the hidden units is exact, but the model defines a rigid, implicit prior. On the other hand, a two hidden layer DBM defines a more flexible prior over the hidden representations, but training and performing inference in a DBM model is considerably harder.

In this paper, we try to find middle ground between these extremes and build a model that combines the best of both. We introduce a two hidden layer DBM model, which we call the Over-Replicated Softmax model. This model is easy to train, has fast approximate inference and still retains some degree of flexibility towards manipulating the prior. Our experiments show that this flexibility is enough to improve significantly on the performance of the standard Replicated Softmax model, both as generative models and as feature extractors even though the new model only has one more parameter than the RBM model. The model also outperforms LDA and DocNADE in terms of classification and retrieval tasks.

2. Over-Replicated Softmax Model

The Over-Replicated Softmax model is a family of two hidden layer Deep Boltzmann Machines (DBM). Let us consider constructing a Boltzmann Machine with two hidden layers for a document containing N words, as shown in Fig. 1. The visible layer $\mathbf{V}$ consists of N softmax units. These units are connected to a binary hidden layer $\mathbf{h}^{(1)}$ with shared weights. The second hidden layer consists of M softmax units represented by $\mathbf{H}^{(\mathbf{2})}$. Similar to $\mathbf{V}$, $\mathbf{H}^{(\mathbf{2})}$ is an $M \times K$ binary matrix with $h^{(2)}_{mk} = 1$ if the m-th hidden softmax unit takes on the k-th value.

The energy of the joint configuration $\{\mathbf{V}, \mathbf{h}^{(1)}, \mathbf{H}^{(\mathbf{2})}\}$ is defined as:

$$E(\mathbf{V}, \mathbf{h}^{(1)}, \mathbf{H}^{(\mathbf{2})}; \boldsymbol{\theta}) = -\sum_{i=1}^{N}\sum_{j=1}^{F}\sum_{k=1}^{K} W^{(1)}_{ijk} h^{(1)}_j v_{ik} \quad (1)$$
$$-\sum_{i'=1}^{M}\sum_{j=1}^{F}\sum_{k=1}^{K} W^{(2)}_{i'jk} h^{(1)}_j h^{(2)}_{i'k} - \sum_{i=1}^{N}\sum_{k=1}^{K} v_{ik} b^{(1)}_{ik}$$
$$-(M+N)\sum_{j=1}^{F} h^{(1)}_j a_j - \sum_{i=1}^{M}\sum_{k=1}^{K} h^{(2)}_{ik} b^{(2)}_{ik}$$

where $\boldsymbol{\theta} = \{\mathbf{W}^{(1)}, \mathbf{W}^{(2)}, \mathbf{a}, \mathbf{b}^{(1)}, \mathbf{b}^{(2)}\}$ are the model parameters.

We create a separate document-specific DBM with as many visible softmax units as there are words in the document. We also fix the number M of the second-layer softmax units across all documents. We ignore the order of the words by making all the first layer softmax units share the same set of weights. Moreover, the first and second layer weights are tied. Thus we have $W^{(1)}_{ijk} = W^{(2)}_{i'jk} = W_{jk}$ and $b^{(1)}_{ik} = b^{(2)}_{i'k} = b_k$. Compared to the standard Replicated Softmax model, this model has more replicated softmaxes (hence the name "Over-Replicated"). Unlike the visible softmaxes, these additional softmaxes are unobserved and constitute a second hidden layer. The energy can be simplified to:

$$E(\mathbf{V}, \mathbf{h}^{(1)}, \mathbf{H}^{(\mathbf{2})}; \boldsymbol{\theta}) = -\sum_{j=1}^{F}\sum_{k=1}^{K} W_{jk} h^{(1)}_j \left(\hat{v}_k + \hat{h}^{(2)}_k\right) \quad (2)$$
$$-\sum_{k=1}^{K}\left(\hat{v}_k + \hat{h}^{(2)}_k\right) b_k - (M+N)\sum_{j=1}^{F} h^{(1)}_j a_j$$

where $\hat{v}_k = \sum_{i=1}^{N} v_{ik}$ denotes the count for the k^{th} word in the input and $\hat{h}^{(2)}_k = \sum_{i=1}^{M} h^{(2)}_{ik}$ denotes the count for the k^{th} "latent" word in the second hidden layer. The joint probability distribution is defined as:

$$P(\mathbf{V}, \mathbf{h}^{(1)}, \mathbf{H}^{(\mathbf{2})}; \boldsymbol{\theta}) = \frac{\exp\left(-E(\mathbf{V}, \mathbf{h}^{(1)}, \mathbf{H}^{(\mathbf{2})}; \boldsymbol{\theta})\right)}{\mathcal{Z}(\boldsymbol{\theta}, N)},$$

A pleasing property of the Over-Replicated Softmax model is that it has exactly the same number of train-

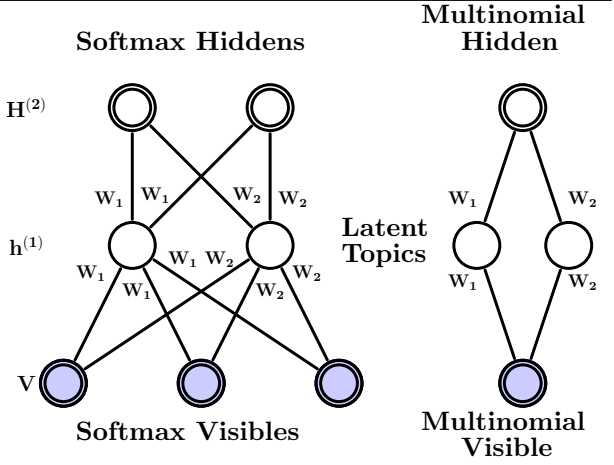


Figure 1. The Over-Replicated Softmax model. The bottom layer represents softmax visible units $\mathbf{V}$. The middle layer represents binary latent topics $\mathbf{h}^{(1)}$. The top layer represents softmax hidden units $\mathbf{H}^{(\mathbf{2})}$. All visible and hidden softmax units share the same set of weights, connecting them to binary hidden units. **Left:** The model for a document containing $N = 3$ words with $M = 2$ softmax hidden units. **Right:** A different interpretation of the model, in which N softmax units with identical weights are replaced by a single multinomial unit which is sampled N times and the M softmax hidden units are replaced by a multinomial unit sampled M times.

able parameters as the Replicated Softmax model. However, the model's marginal distribution over $\mathbf{V}$ is different, as the second hidden layer provides an additional implicit prior. The model's implicit prior over the latent topics $\mathbf{h}^{(1)}$ can be viewed as the geometric mean of the two probability distributions[1]: one defined by an RBM composed of $\mathbf{v}$ and $\mathbf{h}^{(1)}$, and the other defined by an RBM composed of and $\mathbf{h}^{(1)}$ and $\mathbf{H^{(2)}}$:

$$P(\mathbf{h}^{(1)};\boldsymbol{\theta}) = \frac{1}{\mathcal{Z}(\boldsymbol{\theta},N)} \underbrace{\left(\sum_{\mathbf{v}} \exp\left(\sum_{j=1}^{F}\sum_{k=1}^{K} W_{jk}\hat{v}_k h_j^{(1)}\right)\right)}_{\text{RBM with } \mathbf{h}^{(1)} \text{ and } \mathbf{v}} \underbrace{\left(\sum_{\mathbf{H^{(2)}}} \exp\left(\sum_{j=1}^{F}\sum_{k=1}^{K} W_{jk}\hat{h}_k^{(2)} h_j^{(1)}\right)\right)}_{\text{RBM with } \mathbf{h}^{(1)} \text{ and } \mathbf{H^{(2)}}}.$$

Observe that $\sum_{k=1}^{K}\hat{v}_k = N$ and $\sum_{k=1}^{K}\hat{h}_k^{(2)} = M$, so the strength of this prior can be varied by changing the number M of second-layer softmax units. For example, if $M = N$, then the model's marginal distribution over $\mathbf{h}^{(1)}$, defined in Eq. 3, is given by the product of two identical distributions. In this DBM, the second-layer performs 1/2 of the modeling work compared to the first layer (Salakhutdinov & Hinton, 2012). Hence, for documents containing few words ($N \ll M$) the prior over hidden topics $\mathbf{h}^{(1)}$ will be dominated by the second-layer, whereas for long documents ($N \gg M$) the effect of having a second-layer will diminish. As we show in our experimental results, having this additional flexibility in terms of defining an implicit prior over $\mathbf{h}^{(1)}$ significantly improves model performance, particularly for small and medium-sized documents.

2.1. Learning

Let $\mathbf{h} = \{\mathbf{h}^{(1)}, \mathbf{H^{(2)}}\}$ be the set of hidden units in the two-layer DBM. Given a collection of L documents $\{\mathbf{V}\}_{l=1}^{L}$, the derivative of the log-likelihood with respect to model parameters W takes the form:

$$\frac{1}{L}\sum_{l=1}^{L}\frac{\partial \log P(\mathbf{V_l};\boldsymbol{\theta})}{\partial W_{jk}} = \mathbb{E}_{P_{\text{data}}}\left[(\hat{v}_k + \hat{h}_k^{(2)})h_j^{(1)}\right] - \mathbb{E}_{P_{\text{Model}}}\left[(\hat{v}_k + \hat{h}_k^{(2)})h_j^{(1)}\right],$$

where $\mathbb{E}_{P_{\text{data}}}[\cdot]$ denotes an expectation with respect to the data distribution $P_{\text{data}}(\mathbf{h},\mathbf{V}) = P(\mathbf{h}|\mathbf{V};\boldsymbol{\theta})P_{\text{data}}(\mathbf{V})$, with $P_{\text{data}}(\mathbf{V}) = \frac{1}{L}\sum_l \delta(\mathbf{V}-\mathbf{V}_l)$ representing the empirical distribution, and $\mathbb{E}_{P_{\text{Model}}}[\cdot]$ is an expectation with respect to the distribution defined by the model. Similar to the Replicated Softmax model, exact maximum likelihood learning is intractable, but approximate learning can be performed using a variational approach (Salakhutdinov & Hinton, 2009a). We use mean-field inference to estimate data-dependent expectations and an MCMC based stochastic approximation procedure to approximate the models expected sufficient statistics.

Consider any approximating distribution $Q(\mathbf{h}|\mathbf{V};\boldsymbol{\mu})$, parameterized by a vector of parameters $\boldsymbol{\mu}$, for the posterior $P(\mathbf{h}|\mathbf{V};\boldsymbol{\theta})$. Then the log-likelihood of the DBM model has the following variational lower bound:

$$\log P(\mathbf{V};\boldsymbol{\theta}) \geq \sum_{\mathbf{h}} Q(\mathbf{h}|\mathbf{V};\boldsymbol{\mu})\log P(\mathbf{V},\mathbf{h};\boldsymbol{\theta}) + \mathcal{H}(Q), \quad (3)$$

where $\mathcal{H}(\cdot)$ is the entropy functional. The bound becomes tight if and only if $Q(\mathbf{h}|\mathbf{V};\boldsymbol{\mu}) = P(\mathbf{h}|\mathbf{V};\boldsymbol{\theta})$.

For simplicity and speed, we approximate the true posterior $P(\mathbf{h}|\mathbf{V};\boldsymbol{\theta})$ with a fully factorized approximating distribution over the two sets of hidden units, which corresponds to the so-called mean-field approximation:

$$Q^{MF}(\mathbf{h}|\mathbf{V};\boldsymbol{\mu}) = \prod_{j=1}^{F} q(h_j^{(1)}|\mathbf{V}) \prod_{i=1}^{M} q(h_i^{(2)}|\mathbf{V}), \quad (4)$$

where $\boldsymbol{\mu} = \{\boldsymbol{\mu}^{(1)}, \boldsymbol{\mu}^{(2)}\}$ are the mean-field parameters with $q(h_j^{(1)}=1) = \mu_j^{(1)}$ and $q(h_{ik}^{(2)}=1) = \mu_k^{(2)}$, $\forall i \in \{1,\ldots,M\}$, s.t. $\sum_{k=1}^{K}\mu_k^{(2)} = 1$. Note that due to the shared weights across all of the hidden softmaxes, $q(h_{ik}^{(2)})$ does not dependent on i. In this case the variational lower bound on the log-probability of the data takes a particularly simple form:

$$\begin{aligned}\log P(\mathbf{V};\boldsymbol{\theta}) &\geq \sum_{\mathbf{h}} Q^{MF}(\mathbf{h}|\mathbf{V};\boldsymbol{\mu})\log P(\mathbf{V},\mathbf{h};\boldsymbol{\theta}) + \mathcal{H}(Q^{MF}) \\ &\geq \left(\hat{\mathbf{v}}^\top + M\boldsymbol{\mu}^{(2)\top}\right)\mathbf{W}\boldsymbol{\mu}^{(1)} - \log\mathcal{Z}(\boldsymbol{\theta},N) + \mathcal{H}(Q^{MF}), \quad (5)\end{aligned}$$

where $\hat{\mathbf{v}}$ is a $K\times 1$ vector, with its k^{th} element $\hat{v}_k$ containing the count for the k^{th} word. Since $\sum_{k=1}^{K}\hat{v}_k = N$ and $\sum_{k=1}^{K}\mu_k^{(2)} = 1$, the first term in the bound linearly combines the effect of the data (which scales as N) with the prior (which scales as M). For each training example, we maximize this lower bound with respect to the variational parameters $\boldsymbol{\mu}$ for fixed parameters $\boldsymbol{\theta}$, which results in the mean-field fixed-point equations:

$$\mu_j^{(1)} \leftarrow \sigma\left(\sum_{k=1}^{K} W_{jk}\left(\hat{v}_k + M\mu_k^{(2)}\right)\right) \quad (6)$$

$$\mu_k^{(2)} \leftarrow \frac{\exp\left(\sum_{j=1}^{F} W_{jk}\mu_j^{(1)}\right)}{\sum_{k'=1}^{K}\exp\left(\sum_{j=1}^{F} W_{jk'}\mu_j^{(1)}\right)} \quad (7)$$

[1] We omit the bias terms for clarity of presentation.

where $\sigma(x) = 1/(1 + \exp(-x))$ is the logistic function. To solve these fixed-point equations, we simply cycle through layers, updating the mean-field parameters within a single layer.

Given the variational parameters $\boldsymbol{\mu}$, the model parameters $\boldsymbol{\theta}$ are then updated to maximize the variational bound using an MCMC-based stochastic approximation (Salakhutdinov & Hinton, 2009a; Tieleman, 2008; Younes, 2000). Let $\boldsymbol{\theta}_t$ and $\mathbf{x}_t = \{\mathbf{V}_t, \mathbf{h}^{(1)}{}_t, \mathbf{h}^{(2)}{}_t\}$ be the current parameters and the state. Then $\mathbf{x}_t$ and $\boldsymbol{\theta}_t$ are updated sequentially as follows: given $\mathbf{x}_t$, sample a new state $\mathbf{x}_{t+1}$ using alternating Gibbs sampling. A new parameter $\boldsymbol{\theta}_{t+1}$ is then obtained by making a gradient step, where the intractable model's expectation $\mathbb{E}_{P_{\text{model}}}[\cdot]$ in the gradient is replaced by a point estimate at sample $\mathbf{x}_{t+1}$.

In practice, to deal with variable document lengths, we take a minibatch of data and run one Markov chain for each training case for a few steps. To update the model parameters, we use an average over those chains. Similar to Contrastive Divergence learning, in order to provide a good starting point for the sampling, we initialize each chain at $\hat{\mathbf{h}}^{(1)}$ by sampling from the mean-field approximation to the posterior $q(\mathbf{h}^{(1)}|\mathbf{V})$.

2.2. An Efficient Pretraining Algorithm

The proper training procedure for the DBM model described above is quite slow. This makes it very important to pretrain the model so that the model parameters start off in a nice region of space. Fortunately, due to parameter sharing between the visible and hidden softmax units, there exists an efficient pretraining method which makes the proper training almost redundant.

Consider a DBM with N observed and M hidden softmax units. Let us first assume that the number of hidden softmaxes M is the same as the number of words N in a given document. If we were given the initial state vector $\mathbf{H^{(2)}}$, we could train this DBM using one-step contrastive divergence with mean-field reconstructions of both the states of the visible and the hidden softmax units, as shown in Fig. 2. Since we are not given the initial state, one option is to set $\mathbf{H^{(2)}}$ to be equal to the data $\mathbf{V}$. Provided we use mean-field reconstructions for both the visible and second-layer hidden units, one-step contrastive divergence is then exactly the same as training a Replicated Softmax RBM with only one hidden layer but with bottom-up weights that are twice the top-down weights.

To pretrain a DBM with different number of visible and hidden softmaxes, we train an RBM with the bottom-up weights scaled by a factor of $1 + \frac{M}{N}$. In other words, in place of using $\mathbf{W}$ to compute the conditional probability of the hidden units (see Eq. **??**), we use $(1 + \frac{M}{N})\mathbf{W}$:

$$P(h_j^{(1)} = 1|\mathbf{V}) = \sigma\left((1 + \frac{M}{N})\sum_{k=1}^{K} v_k W_{kj}\right). \qquad (8)$$

The conditional probability of the observed softmax units remains the same as in Eq. **??**. This procedure is equivalent to training an RBM with $N+M$ observed visible units with each of the M extra units set to be the empirical word distribution in the document, i.e.. for $i \in \{N+1, \ldots, N+M\}$,

$$v_{ik} = \frac{\sum_{j=1}^{N} v_{jk}}{\sum_{j=1}^{N}\sum_{k'=1}^{K} v_{jk'}}$$

Thus the M extra units are not 1-of-K, but represent distributions over the K words[2].

This way of pretraining the Over-Replicated Softmax DBMs with tied weights will not in general maximize the likelihood of the weights. However, in practice it produces models that reconstruct the training data well and serve as a good starting point for generative fine-tuning of the two-layer model.

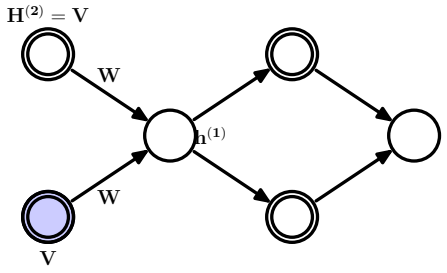


Figure 2. Pretraining a two-layer Boltzmann Machine using one-step contrastive divergence. The second hidden softmax layer is initialized to be the same as the observed data. The units in the first hidden layer have stochastic binary states, but the reconstructions of both the visible and second hidden layer use probabilities, so both reconstructions are identical.

2.3. Inference

The posterior distribution $P(\mathbf{h}^{(1)}|\mathbf{V})$ represents the latent topic structure of the observed document. Conditioned on the document, these activation probabilities can be inferred using the mean-field approximation used to infer data-dependent statistics during training.

[2]Note that when $M = N$, we recover the setting of having the bottom-up weights being twice the top-down weights.

A fast alternative to the mean-field posterior is to multiply the visible to hidden weights by a factor of $1+\frac{M}{N}$ and approximate the true posterior with a single matrix multiply, using Eq. 8. Setting $M = 0$ recovers the proper posterior inference step for the standard Replicated Softmax model. This simple scaling operation leads to significant improvements. The results reported for retrieval and classification experiments used the fast pretraining and fast inference methods.

2.4. Choosing M

The number of hidden softmaxes M affects the strength of the additional prior. The value of M can be chosen using a validation set. Since the value of M is fixed for all Over-Replicated DBMs, the effect of the prior will be less for documents containing many words. This is particularly easy to see in Eq. 8. As N becomes large, the scaling factor approaches 1, diminishing the part of implicit prior coming from the M hidden softmax units. Thus the value of M can be chosen based on the distribution of lengths of documents in the corpus.

3. Experiments

In this section, we evaluate the Over-Replicated Softmax model both as a generative model and as a feature extraction method for retrieval and classification. Two datasets are used - 20 Newsgroups and Reuters Corpus Volume I (RCV1-v2).

3.1. Description of datasets

The 20 Newsgroups dataset consists of 18,845 posts taken from the Usenet newsgroup collection. Each post belongs to exactly one newsgroup. Following the preprocessing in (Salakhutdinov & Hinton, 2009b) and (Larochelle & Lauly, 2012), the data was partitioned chronologically into 11,314 training and 7,531 test articles. After removing stopwords and stemming, the 2000 most frequent words in the training set were used to represent the documents.

The Reuters RCV1-v2 contains 804,414 newswire articles. There are 103 topics which form a tree hierarchy. Thus documents typically have multiple labels. The data was randomly split into 794,414 training and 10,000 test cases. The available data was already preprocessed by removing common stopwords and stemming. We use a vocabulary of the 10,000 most frequent words in the training dataset.

3.2. Perplexity

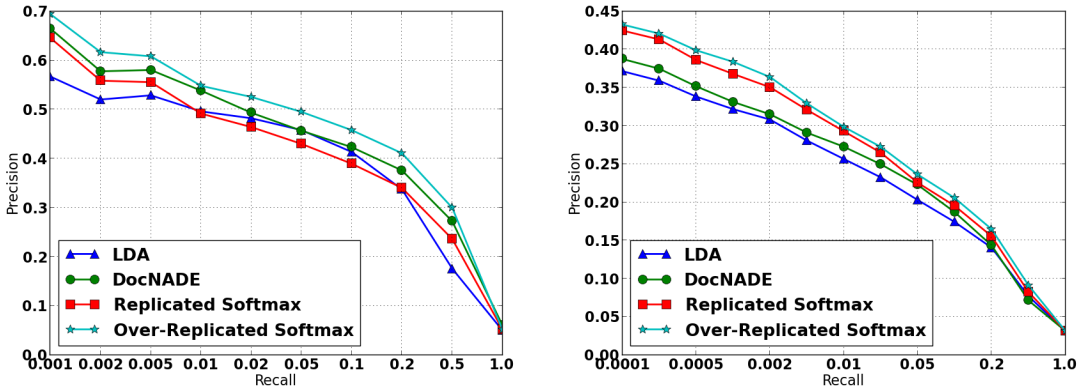
We compare the Over-Replicated Softmax model with the Replicated Softmax model in terms of perplexity. Computing perplexities involves computing the partition functions for these models. We used Annealed Importance Sampling (Neal, 2001) for doing this. In order to get reliable estimates, we ran 128 Markov chains for each document length. The average test perplexity per word was computed as $\exp\left(-1/L\sum_{l=1}^{L} 1/N_l \log p(\mathbf{v}_l)\right)$, where N_l is the number of words in document l. Table 1 shows the perplexity averaged over $L = 1000$ randomly chosen test cases for each data set. Each of the models has 128 latent topics. Table 1 shows that the Over-Replicated Softmax model assigns slightly lower perplexity to the test data compared to the Replicated Softmax model. For the Reuters data set the perplexity decreases from 1081 to 1060, and for 20 Newsgroups, it decreases from 965 to 958. Though the decrease is small, it is statistically significant since the standard deviation was typically ± 2 over 10 random choices of 1000 test cases. Increasing the value of M increases the strength of the prior, which leads to further improvements in perplexities. Note that the estimate of the log probability for 2-layered Boltzmann Machines is a lower bound on the actual log probability. So the perplexities we show are upper bounds and the actual perplexities may be lower (provided the estimate of the partition function is close to the actual value).

Table 1. Comparison of the average test perplexity per word. All models use 128 topics.

	20 News	**Reuters**
Training set size	11,072	794,414
Test set size	7,052	10,000
Vocabulary size	2,000	10,000
Avg Document Length	51.8	94.6
Perplexities		
Unigram	1335	2208
Replicated Softmax	965	1081
Over-Rep. Softmax ($M = 50$)	961	1076
Over-Rep. Softmax ($M = 100$)	**958**	**1060**

3.3. Document Retrieval

In order to do retrieval, we represent each document $\mathbf{V}$ as the conditional posterior distribution $P(\mathbf{h}^{(1)}|\mathbf{V})$. This can be done exactly for the Replicated Softmax and DocNADE models. For two-layered Boltzmann Machines, we extract this representation using the fast approximate inference as described in Sec. 2.3. Performing more accurate inference using the mean-field approximation method did not lead to statistically different results. For the LDA, we used 1000 Gibbs sweeps per test document in order to get an approximate posterior over the topics.



(a) 20 Newsgroups (b) Reuters

Figure 3. Comparison of Precision-Recall curves for document retrieval. All models use 512 hidden units. All Over-Replicated Softmax models use $M = 100$ latent words.

Documents in the training set (including the validation set) were used as a database. The test set was used as queries. For each query, documents in the database were ranked using cosine distance as the similarity metric. The retrieval task was performed separately for each label and the results were averaged. Fig. 3 compares the precision-recall curves. As shown by Fig. 3, the Over-Replicated Softmax DBM outperforms other models on both datasets, particularly when retrieving the top few documents.

3.4. Document Classification

In this set of experiments, we evaluate the learned representations from the Over-Replicated Softmax model for the purpose of document classification. Since the objective is to evaluate the quality of the representation, simple linear classifiers were used. Multinomial logistic regression with a cross entropy loss function was used for the 20 newsgroups data set. The evaluation metric was classification accuracy. For the Reuters dataset, we used independent logistic regressions for each label since it is a multi-label classification problem. The evaluation metric was Mean Average Precision.

Table 2 shows the results of these experiments. The Over-Replicated Softmax model performs significantly better than the standard Replicated Softmax model and LDA across different network sizes on both datasets. For the 20 newsgroups dataset using 512 topics, LDA gets 64.2% accuracy. Replicated Softmax (67.7%) and DocNADE (68.4%) improve upon this. The Over-Replicated Softmax model further improves the result to 69.4%. The difference is larger for the Reuters dataset. In terms of Mean Average Precision (MAP), the Over-Replicated Softmax model achieves 0.453 which is a very significant improvement upon DocNADE (0.427) and Replicated Softmax (0.421).

Table 2. Comparison of Classification accuracy on 20 Newsgroups dataset and Mean Average Precision on Reuters RCV1-v2.

Model	**20 News**		**Reuters**	
	128	**512**	**128**	**512**
LDA	65.7	64.2	0.304	0.351
DocNADE	**67.0**	68.4	0.388	0.417
Replicated Softmax	65.9	67.7	0.390	0.421
Over-Rep. Softmax	66.8	**69.1**	**0.401**	**0.453**

We further examined the source of improvement by analyzing the effect of document length on the classification performance. Similar to retrieval, we found that the Over-Replicated Softmax model performs well on short documents. For long documents, the performance of the different models was similar.

4. Conclusion

The Over-Replicated Softmax model described in this paper is an effective way of defining a flexible prior over the latent topic features of an RBM. This model causes no increase in the number of trainable parameters and only a minor increase in training algorithm complexity. Deep Boltzmann Machines are typically slow to train. However, using a fast approximate inference it is possible to train the model with CD, just like an RBM. The features extracted from documents using the Over-Replicated Softmax model perform better than features from the standard Replicated Softmax and LDA models and are comparable to DocNADE across different network sizes.

References

Blei, David M. Probabilistic topic models. *Commun. ACM*, 55(4):77–84, 2012.

Blei, David M., Ng, Andrew, and Jordan, Michael. Latent dirichlet allocation. *JMLR*, 3:993–1022, 2003.

Blei, David M., Griffiths, Thomas L., and Jordan, Michael I. The nested chinese restaurant process and bayesian nonparametric inference of topic hierarchies. *J. ACM*, 57(2), 2010.

Canini, K., Shi, L., and Griffiths, T. Online inference of topics with latent Dirichlet allocation. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, volume 5, 2009.

Griffiths, T. and Steyvers, M. Finding scientific topics. In *Proceedings of the National Academy of Sciences*, volume 101, pp. 5228–5235, 2004.

Larochelle, Hugo and Lauly, Stanislas. A neural autoregressive topic model. In *Advances in Neural Information Processing Systems 25*, pp. 2717–2725. 2012. URL http://books.nips.cc/papers/files/nips25/NIPS2012_1253.pdf.

Mimno, D. and McCallum, A. Topic models conditioned on arbitrary features with dirichlet-multinomial regression. In *UAI*, pp. 411–418, 2008.

Neal, Radford M. Annealed importance sampling. *Statistics and Computing*, 11(2):125–139, April 2001. ISSN 0960-3174. doi: 10.1023/A:1008923215028. URL http://dx.doi.org/10.1023/A:1008923215028.

Salakhutdinov, R. R. and Hinton, G. E. Deep Boltzmann machines. In *Proceedings of the International Conference on Artificial Intelligence and Statistics*, volume 12, 2009a.

Salakhutdinov, Ruslan and Hinton, Geoff. A better way to pretrain deep boltzmann machines. In *Advances in Neural Information Processing Systems 25*, pp. 2456–2464. 2012. URL http://books.nips.cc/papers/files/nips25/NIPS2012_1178.pdf.

Salakhutdinov, Ruslan and Hinton, Geoffrey. Replicated softmax: an undirected topic model. In *Advances in Neural Information Processing Systems 22*, pp. 1607–1614. 2009b.

Teh, Y. W., Jordan, M. I., Beal, M. J., and Blei, D. M. Hierarchical Dirichlet processes. *Journal of the American Statistical Association*, 101(476):1566–1581, 2006.

Teh, Y. W., Kurihara, K., and Welling, M. Collapsed variational inference for HDP. In *Advances in Neural Information Processing Systems*, volume 20, 2008.

Tieleman, T. Training restricted Boltzmann machines using approximations to the likelihood gradient. In *ICML*. ACM, 2008.

Wang, Chong and Blei, David M. Variational inference for the nested chinese restaurant process. In *NIPS*, pp. 1990–1998, 2009.

Younes, L. On the convergence of Markovian stochastic algorithms with rapidly decreasing ergodicity rates, March 17 2000.