Topic Modeling of Multimodal Data: an Autoregressive Approach*

Yin Zheng Yu-Jin Zhang
Tsinghua National Laboratory for Information Science
and Technology, Department of Electronic Engineering
Tsinghua University, Beijing, China, 100084
y-zheng09@mails.tsinghua.edu.cn

Hugo Larochelle
Département d'Informatique
Université de Sherbrooke
Sherbrooke (QC), Canada, J1K 2R1
hugo.larochelle@usherbrooke.ca

Abstract

Topic modeling based on latent Dirichlet allocation (LDA) has been a framework of choice to deal with multimodal data, such as in image annotation tasks. Recently, a new type of topic model called the Document Neural Autoregressive Distribution Estimator (DocNADE) was proposed and demonstrated state-of-the-art performance for text document modeling. In this work, we show how to successfully apply and extend this model to multimodal data, such as simultaneous image classification and annotation. Specifically, we propose SupDocNADE, a supervised extension of DocNADE, that increases the discriminative power of the hidden topic features by incorporating label information into the training objective of the model and show how to employ SupDocNADE to learn a joint representation from image visual words, annotation words and class label information. We also describe how to leverage information about the spatial position of the visual words for SupDocNADE to achieve better performance in a simple, yet effective manner. We test our model on the LabelMe and UIUC-Sports datasets and show that it compares favorably to other topic models such as the supervised variant of LDA and a Spatial Matching Pyramid (SPM) approach.

1. Introduction

Multimodal data modeling, which combines information from different sources, is increasingly attracting attention in computer vision, such as in [1, 2, 3, 4, 5, 6, 7]. Work on multimodal data is often based on a topic model, the most popular being latent Dirichlet allocation or LDA [8]. LDA is a generative model for documents that originates from the natural language processing community, but has had great success in computer vision [8, 9]. LDA models a document as a multinomial distribution over topics, where a topic is itself a multinomial distribution over words. While the distribution over topics is specific for each document, the topic-dependent distributions over words are shared across all documents. Topic models can thus extract a meaningful, semantic representation from a document by inferring its latent distribution over topics from the words it contains. In the context of computer vision, LDA can be used by first extracting so-called "visual words" from images, convert the images into visual word documents and training an LDA topic model on the bags of visual words.

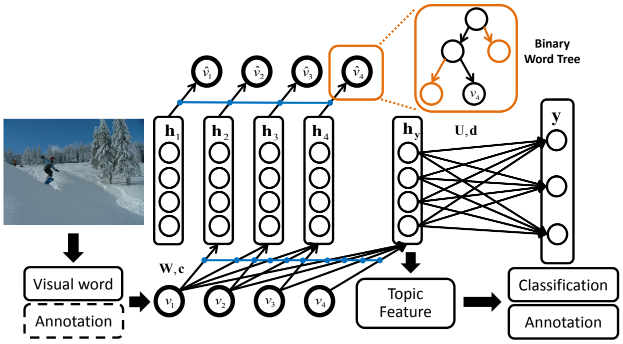


Figure 1. Illustration of SupDocNADE for multimodal image data. Visual words, annotation words and class label y are modeled as $p(\mathbf{v}, y) = p(y|\mathbf{v}) \prod_i p(v_i|v_1, \ldots, v_{i-1})$. All conditionals $p(y|\mathbf{v})$ and $p(v_i|v_1, \ldots, v_{i-1})$ are modeled using neural networks with shared weights. Each predictive word conditional $p(v_i|v_1, \ldots, v_{i-1})$ (noted $\hat{v}_i$ for brevity) follows a tree decomposition where each leaf is a possible word. At test time, the annotation words are not used (illustrated with a dotted box) to compute the image's topic feature representation.

To deal with multimodal data, some variants of LDA have been proposed recently [2, 5, 4, 9]. For instance, Correspondence LDA (Corr-LDA) [2] was proposed to discover the relationship between images and annotation modalities, by assuming each image topic must have a correspond-

[*] This work was partially supported by the Natural Sciences and Engineering Research Council of Canada, the National Nature Science Foundation of China (NNSF: 61171118) and Specialized Research Fund for the Doctoral Program of Higher Education in China (SRFDP-20110002110057).

ing text topic. Multimodal LDA [5] generalizes Corr-LDA by learning a regression module relating the topics from the different modalities. Multimodal Document Random Field Model (MDRF) [4] was also proposed to deal with multimodal data, which learns cross-modality similarities from a document corpus containing multinomial data. Besides the annotation words, the class label modality can also be embedded into LDA, such as in supervised LDA (sLDA) [10, 9]. By modeling the image visual words, annotation words and their class labels, the discriminative power of the learned image representations could thus be improved.

At the heart of most topic models is a generative story in which the image's latent representation is generated first and the visual words are subsequently produced from this representation. The appeal of this approach is that the task of extracting the representation from observations is easily framed as a probabilistic inference problem, for which many general purpose solutions exist. The disadvantage however is that as a model becomes more sophisticated, inference becomes less trivial and more computationally expensive. In LDA for instance, inference of the distribution over topics does not have a closed-form solution and must be approximated, either using variational approximate inference or MCMC sampling. Yet, the model is actually relatively simple, making certain simplifying independence assumptions such as the conditional independence of the visual words given the image's latent distribution over topics.

Recently, an alternative generative modeling approach for documents was proposed by Larochelle and Lauly [11]. Their model, the Document Neural Autoregressive Distribution Estimator (DocNADE), models directly the joint distribution of the words in a document, by decomposing it as a product of conditional distributions (through the probability chain rule) and modeling each conditional using a neural network. Hence, DocNADE doesn't incorporate any latent random variables over which potentially expensive inference must be performed. Instead, a document representation can be computed efficiently in a simple feed-forward fashion, using the value of the neural network's hidden layer. Larochelle and Lauly [11] also show that DocNADE is a better generative model of text documents and can extract a useful representation for text information retrieval.

In this paper, we consider the application of DocNADE to deal with multimodal data in computer vision. More specifically, we propose a supervised variant of DocNADE (SupDocNADE), which can be used to model the joint distribution over an image's visual words, annotation words and class label. The model is illustrated in Figure 1. We investigate how to successfully incorporate spatial information about the visual words and highlight the importance of calibrating the generative and discriminative components of the training objective. Our results confirm that this approach can outperform other topic models, such as the supervised variant of LDA.

2. Related Work

Multimodal data is often modeled using extensions of the basic LDA topic model, such as Corr-LDA [2], Multimodal LDA [5] and MDRF [4]. In this paper, we focus on learning a joint representation from three different modalities: *image visual words, annotations*, and *class labels*. The class label describes the image globally with a single descriptive label (such as *coast*, *outdoor*, *inside city*, etc.), while the annotation focuses on tagging the local content within the image. Wang et al. [9] proposed a supervised LDA formulation to tackle this problem. Wang et al. [12] opted instead for a maximum margin formulation of LDA (MMLDA). Our work also belongs to this line of work, extending topic models to a supervised variant: our contribution is thus to extend a different topic model, DocNADE, to this context for multimodal data modeling.

What distinguishes DocNADE from other topic models is its reliance on an autoregressive neural network architecture. Neural networks are increasingly used for the probabilistic modeling of images (see [13] for a review). In the realm of document modeling, Salakhutdinov and Hinton [14] proposed a Replicated Softmax model for bags of words. DocNADE is in fact inspired by that model and was shown to improve over its performance while being much more computationally efficient. There are also some other neural network based approaches for multimodal data modeling, such as [15, 16]. One advantage of DocNADE over these methods is that it can be trained efficiently without approximation or sampling.

3. Document NADE

In this section, we describe the original DocNADE model. In Larochelle and Lauly [11], DocNADE was used to model documents of real words, belonging to some predefined vocabulary. To model image data, we assume that images have first been converted into a bag of visual words. A standard approach is to learn a vocabulary of visual words by performing K-means clustering on SIFT descriptors densely exacted from all training images. See Section 5.2 for more details about this procedure. From that point on, any image can thus be represented as a bag of visual words $\mathbf{v} = [v_1, v_2, \ldots, v_D]$, where each v_i is the index of the closest K-means cluster to the i^{th} SIFT descriptor extracted from the image and D is the number of extracted descriptors.

DocNADE models the joint probability of the visual words $p(\mathbf{v})$ by rewritting it as

$$p\left(\mathbf{v}\right) = \prod_{i=1}^{D} p\left(v_i|\mathbf{v}_{<i}\right) \tag{1}$$

and modeling instead each conditional $p(v_i|\mathbf{v}_{<i})$, where $\mathbf{v}_{<i}$ is the subvector containing all v_j such that $j < i$. Notice that Equation 1 is true for any distribution, based on the probability chain rule. Hence, the main assumption made by DocNADE is in the form of the conditionals. Specifically, DocNADE assumes that each conditional can be modeled and learned by a feedforward neural network.

One possibility would be to model $p(v_i|\mathbf{v}_{<i})$ with the following architecture:

$$\mathbf{h}_i\left(\mathbf{v}_{<i}\right) = \mathbf{g}\left(\mathbf{c} + \sum_{k<i} \mathbf{W}_{:,v_k}\right) \tag{2}$$

$$p\left(v_i = w|\mathbf{v}_{<i}\right) = \frac{\exp\left(b_w + \mathbf{V}_{w,:}\mathbf{h}_i\left(\mathbf{v}_{<i}\right)\right)}{\sum_{w'} \exp\left(b_{w'} + \mathbf{V}_{w',:}\mathbf{h}_i\left(\mathbf{v}_{<i}\right)\right)} \tag{3}$$

where $g(\cdot)$ is an element-wise non-linear activation function, $\mathbf{W} \in \mathbb{R}^{H \times K}$ and $\mathbf{V} \in \mathbb{R}^{K \times H}$ are the connection parameter matrices, $\mathbf{c} \in \mathbb{R}^N$ and $\mathbf{b} \in \mathbb{R}^K$ are bias parameter vectors and H, K are the number of hidden units (topics) and vocabulary size, respectively.

Computing the distribution $p(v_i = w|\mathbf{v}_{<i})$ of Equation 3 requires time linear in K. In practice, this is too expensive, since it must be computed for each of the D visual words v_i. To address this issue, Larochelle and Lauly [11] propose to use a balanced binary tree to decompose the computation of the conditionals and obtain a complexity logarithmic in K. This is achieved by randomly assigning all visual words to a different leaf in a binary tree. Given this tree, the probability of a word is modeled as the probability of reaching its associated leaf from the root. Larochelle and Lauly [11] model each left/right transition probabilities in the binary tree using a set of binary logistic regressors taking the hidden layer $\mathbf{h}_i(\mathbf{v}_{<i})$ as input. The probability of a given word can then be obtained by multiplying the probabilities of each left/right choices of the associated tree path.

Specifically, let $\mathbf{l}\left(v_i\right)$ be the sequence of tree nodes on the path from the root to the leaf of v_i and let $\pi\left(v_i\right)$ be the sequence of binary left/right choices at the internal nodes along that path. For example, $l\left(v_i\right)_1$ will always be the root node of the binary tree, and $\pi\left(v_i\right)_1$ will be 0 if the word leaf v_i is in the left subtree or 1 otherwise. Let $\mathbf{V} \in \mathbb{R}^{T \times H}$ now be the matrix containing the logistic regression weights and $\mathbf{b} \in \mathbb{R}^T$ be a vector containing the biases, where T is the number of inner nodes in the binary tree and H is the number of hidden units. The probability $p(v_i = w|\mathbf{v}_{<i})$ is now modeled as

$$p(v_i = w|\mathbf{v}_{<i}) = \prod_{k=1}^{|\pi(v_i)|} p(\pi\left(v_i\right)_k|\mathbf{v}_{<i}) \ , \tag{4}$$

where

$$p(\pi\left(v_i\right)_k = 1|\mathbf{v}_{<i}) = \text{ sigm}\left(b_{l(v_i)_m} + \mathbf{V}_{l(v_i)_m,:}\mathbf{h}_i\left(\mathbf{v}_{<i}\right)\right) \tag{5}$$

are the internal node logistic regression outputs and $\text{sigm}(x) = 1/(1 + \exp(-x))$ is the sigmoid function. By using a balanced tree, we are guaranteed that computing Equation 4 involves only $O(\log K)$ logistic regression outputs. One could attempt to optimize the organization of the words within the tree, but a random assignment of the words to leaves works well in practice [11].

Thus, by combining Equations 2, 4 and 5, we can compute the probability $p\left(\mathbf{v}\right) = \prod_{i=1} p\left(v_i|\mathbf{v}_{<i}\right)$ for any document under DocNADE. To train the parameters $\theta = \{\mathbf{W}, \mathbf{V}, \mathbf{b}, \mathbf{c}\}$ of DocNADE, we simply optimize the average negative log-likelihood of the training set documents using stochastic gradient descent.

Equations 4,5 indicate that the conditional probability of each word v_i requires computing the position dependent hidden layer $\mathbf{h}_i\left(\mathbf{v}_{<i}\right)$, which extracts a representation out of the bag of previous visual words $\mathbf{v}_{<i}$. Since computing $\mathbf{h}_i\left(\mathbf{v}_{<i}\right)$ is in $O(HD)$ on average, and there are D hidden layers $\mathbf{h}_i\left(\mathbf{v}_{<i}\right)$ to compute, then a naive procedure for computing all hidden layers would be in $O(HD^2)$.

However, noticing that

$$\begin{aligned} \mathbf{h}_{i+1}\left(\mathbf{v}_{<i+1}\right) &= \mathbf{g}\left(\mathbf{c} + \sum_{k<i+1} \mathbf{W}_{:,v_k}\right) & (6) \\ &= \mathbf{g}\left(\mathbf{W}_{:,v_i} + \mathbf{c} + \sum_{k<i} \mathbf{W}_{:,v_k}\right) & (7) \end{aligned}$$

and exploiting that fact that the weight matrix $\mathbf{W}$ is the same across all conditionals, the linear transformation $\mathbf{c} + \sum_{k<i} \mathbf{W}_{:,v_k}$ can be reused from the computation of the previous hidden layer $\mathbf{h}_i(\mathbf{v}_{<i})$ to compute $\mathbf{h}_{i+1}(\mathbf{v}_{<i+1})$. With this procedure, computing all hidden layers $\mathbf{h}_i(\mathbf{v}_{<i})$ sequentially from $i = 1$ to $i = D$ becomes in $O(HD)$.

Finally, since the computation complexity of each of the $O(\log K)$ logistic regressions in Equation 4 is $O(H)$, the total complexity of computing $p(v_i = w|\mathbf{v}_{<i})$ is $O(\log(K)HD)$. In practice, the length of document D and the number of hidden units H tends to be small, while $\log(K)$ will be small even for large vocabularies. Thus DocNADE can be used and trained efficiently.

Once the model is trained, a latent representation can be extracted from a new document $\mathbf{v}^*$ as follows:

$$\mathbf{h}_y\left(\mathbf{v}^*\right) = \mathbf{g}\left(\mathbf{c} + \sum_{i}^{D} \mathbf{W}_{:,v_i^*}\right) \ . \tag{8}$$

This representation could be fed to a standard classifier to perform any supervised computer vision task. The index y is used to highlight that it is the representation used to predict the class label y of the image.

4. SupDocNADE for Multimodal Data

In this section, we describe the approach of this paper, inspired by DocNADE, to learn jointly from multimodal data. First, we describe a supervised extension of DocNADE (SupDocNADE), which incorporates the class label modality into training to learn more discriminative hidden features for classification. Then we describe how we exploit the spatial position information of the visual words. Finally, we describe how to jointly model the text annotation modality with SupDocNADE.

4.1. Supervised DocNADE

It has been observed that learning image feature representations using unsupervised topic models such as LDA can perform worse than training a classifier directly on the visual words themselves, using an appropriate kernel such as a pyramid kernel [17]. One reason is that the unsupervised topic features are trained to explain as much of the entire statistical structure of images as possible and might not model well the particular discriminative structure we are after in our computer vision task. This issue has been addressed in the literature by devising supervised variants of LDA, such as Supervised LDA or sLDA [10]. DocNADE also being an unsupervised topic model, we propose here a supervised variant of DocNADE, SupDocNADE, in an attempt to make the learned image representation more discriminative for the purpose of image classification.

Specifically, given an image $\mathbf{v} = [v_1, v_2, \ldots, v_D]$ and its class label $y \in \{1, \ldots, C\}$, SupDocNADE models the full joint distribution as

$$p(\mathbf{v}, y) = p(y|\mathbf{v}) \prod_{i=1}^{D} p\left(v_i|\mathbf{v}_{<i}\right) \quad . \tag{9}$$

As in DocNADE, each conditional is modeled by a neural network. We use the same architecture for $p\left(v_i|\mathbf{v}_{<i}\right)$ as in regular DocNADE. We now only need to define the model for $p(y|\mathbf{v})$.

Since $\mathbf{h}_y\left(\mathbf{v}\right)$ is the image representation that we'll use to perform classification, we propose to model $p\left(y|\mathbf{v}\right)$ as a multiclass logistic regression output computed from $\mathbf{h}_y\left(\mathbf{v}\right)$:

$$p\left(y|\mathbf{v}\right) = \operatorname{softmax}\left(\mathbf{d} + \mathbf{U}\mathbf{h}_y\left(\mathbf{v}\right)\right)_y \tag{10}$$

where $\operatorname{softmax}(\mathbf{a})_i = \exp(a_i)/\sum_{j=1}^{C}\exp(a_j)$, $\mathbf{d} \in \mathbb{R}^C$ is the bias parameter vector in the supervised layer and $\mathbf{U} \in \mathbb{R}^{C\times H}$ is the connection matrix between hidden layer $\mathbf{h}_y$ and the class label.

Put differently, $p\left(y|\mathbf{v}\right)$ is modeled as a regular multiclass neural network, taking as input the bag of visual words $\mathbf{v}$. The crucial difference however with a regular neural network is that some of its parameters (namely the hidden unit parameters $\mathbf{W}$ and $\mathbf{c}$) are also used to model the visual word conditionals $p\left(v_i|\mathbf{v}_{<i}\right)$.

Maximum likelihood training of this model is performed by minimizing the negative log-likelihood

$$-\log p\left(\mathbf{v}, y\right) = -\log p\left(y|\mathbf{v}\right) + \sum_{i=1}^{D} -\log p(v_i|\mathbf{v}_{<i}) \tag{11}$$

averaged over all training images. This is known as generative learning [18]. The first term is a purely discriminative term, while the second is unsupervised and can be understood as a regularizer, that encourages a solution which also explains the unsupervised statistical structure within the visual words. In practice, this regularizer can bias the solution too strongly away from a more discriminative solution that generalizes well. Hence, similarly to previous work on hybrid generative/discriminative learning, we propose instead to weight the importance of the generative term

$$-\log p\left(\mathbf{v}, y\right) = -\log p\left(y|\mathbf{v}\right) + \lambda \sum_{i=1}^{D} -\log p(v_i|\mathbf{v}_{<i}) \tag{12}$$

where λ is treated as a regularization hyper-parameter.

Optimizing the training set average of Equation 12 is performed by stochastic gradient descent, using backpropagation to compute the parameter derivatives. As in regular DocNADE, computation of the training objective and its gradient requires that we define an ordering of the visual words. Though we could have defined an arbitrary path across the image to order the words (e.g. from left to right, top to bottom in the image), we follow Larochelle and Lauly [11] and randomly permute the words before every stochastic gradient update. The implication is that the model is effectively trained to be a good inference model of *any* conditional $p(v_i|\mathbf{v}_{<i})$, for any ordering of the words in $\mathbf{v}$. This again helps fighting against overfitting and better regularizes our model. One could thus think of SupDocNADE as learning from a sequence of *random* fixations performed in a visual scene.

In our experiments, we used the rectified linear function as the activation function

$$\mathbf{g}(\mathbf{a}) = \max(0, \mathbf{a}) = [\max(0, a_1), \ldots, \max(0, a_H)] \tag{13}$$

which often outperforms other activation functions [19] and has been shown to work well for image data [20]. Since this is a piece-wise linear function, the (sub-)gradient with respect to its input, needed by backpropagation to compute the parameter gradients, is simply

$$\mathbf{1}_{(\mathbf{g}(\mathbf{a})>0)} = [1_{(g(a_1)>0)}, \ldots, 1_{(g(a_H)>0)}] \tag{14}$$

where 1_P is 1 if P is true and 0 otherwise.

Algorithms 1 and 2 give pseudocodes for efficiently computing the joint distribution $p\left(\mathbf{v}, y\right)$ and the parameter

Algorithm 1 Computing $p(\mathbf{v}, y)$ using SupDocNADE

```
Input: bag of words representation $\mathbf{v}$, target $y$
Output: $p(\mathbf{v}, y)$
$\mathbf{act} \leftarrow \mathbf{c}$
$p(\mathbf{v}) \leftarrow 1$
for $i$ from 1 to $D$ do
    $\mathbf{h}_i \leftarrow \mathbf{g}(\mathbf{act})$
    $p(v_i|\mathbf{v}_{<i}) = 1$
    for $m$ from 1 to $|\pi(v_i)|$ do
        $p(v_i|\mathbf{v}_{<i}) \leftarrow p(v_i|\mathbf{v}_{<i})\, p(\pi(v_i)_m|\mathbf{v}_{<i})$
    end for
    $p(\mathbf{v}) \leftarrow p(\mathbf{v})\, p(v_i|\mathbf{v}_{<i})$
    $\mathbf{act} \leftarrow \mathbf{act} + \mathbf{W}_{:,v_i}$
end for
$\mathbf{h}^c(\mathbf{v}) \leftarrow \max(0, \mathbf{act})$
$p(y|\mathbf{v}) \leftarrow \mathrm{softmax}(\mathbf{d} + \mathbf{U}\mathbf{h}^c(\mathbf{v}))|_y$
$p(\mathbf{v}, y) \leftarrow p(\mathbf{v})\, p(y|\mathbf{v})$
```

Algorithm 2 Computing SupDocNADE training gradients

```
Input: training vector $\mathbf{v}$, target $y$,
       unsupervised learning weight $\lambda$
Output: gradients of Equation 12 w.r.t. parameters
$f(\mathbf{v}) \leftarrow \mathrm{softmax}(\mathbf{d} + \mathbf{U}\mathbf{h}^c(\mathbf{v}))$
$\delta\mathbf{d} \leftarrow (f(\mathbf{v}) - 1_y)$
$\delta\mathbf{act} \leftarrow (\mathbf{U}^\intercal \delta\mathbf{d}) \circ 1_{\mathbf{h}_y > 0}$
$\delta\mathbf{U} \leftarrow \delta\mathbf{d}\, \mathbf{h}^{c\intercal}$
$\delta\mathbf{c} \leftarrow 0$, $\delta\mathbf{b} \leftarrow 0$, $\delta\mathbf{V} \leftarrow 0$, $\delta\mathbf{W} \leftarrow 0$
for $i$ from $D$ to 1 do
    $\delta\mathbf{h}_i \leftarrow 0$
    for $m$ from 1 to $|\pi(v_i)|$ do
        $\delta t \leftarrow \lambda(p(\pi(v_i)_m|\mathbf{v}_{<i}) - \pi(v_i)_m)$
        $\delta b_{l(v_i)_m} \leftarrow \delta b_{l(v_i)_m} + \delta t$
        $\delta\mathbf{V}_{l(v_i)_m,:} \leftarrow \delta\mathbf{V}_{l(v_i)_m,:} + \delta t\, \mathbf{h}_i^\intercal$
        $\delta\mathbf{h}_i \leftarrow \delta\mathbf{h}_i + \delta t\, \mathbf{V}_{l(v_i)_m,:}^\intercal$
    end for
    $\delta\mathbf{act} \leftarrow \delta\mathbf{act} + \delta\mathbf{h}_i \circ 1_{\mathbf{h}_i > 0}$
    $\delta\mathbf{c} \leftarrow \delta\mathbf{c} + \delta\mathbf{h}_i \circ 1_{\mathbf{h}_i > 0}$
    $\delta\mathbf{W}_{:,v_i} \leftarrow \delta\mathbf{W}_{:,v_i} + \delta\mathbf{act}$
end for
```

gradients of Equation 12 required for stochastic gradient descent training.

4.2. Dealing with Multiple Regions

Spatial information plays an important role for understanding an image. For example, the sky will often appear on the top part of the image, while a car will most often appear at the bottom. A lot of previous work has exploited this intuition successfully. For example, in the seminal work on spatial pyramids [17], it is shown that extracting different visual word histograms over distinct regions instead of a single image-wide histogram can yield substantial gains in performance.

We follow a similar approach, whereby we model both the presence of the visual words and the identity of the region they appear in. Specifically, let's assume the image is divided into several distinct regions $\mathcal{R} = \{R_1, R_2, \ldots, R_M\}$, where M is the number of regions. The image can now be represented as

$$\begin{aligned} \mathbf{v}^{\mathcal{R}} &= [v_1^{\mathcal{R}}, v_2^{\mathcal{R}}, \ldots, v_D^{\mathcal{R}}] \\ &= [(v_1, r_1), (v_2, r_2), \ldots, (v_D, r_D)] \end{aligned} \qquad (15)$$

where $r_i \in \mathcal{R}$ is the region from which the visual word v_i was extracted. To model the joint distribution over these visual words, we decompose it as $p(\mathbf{v}^{\mathcal{R}}) = \prod_i p((v_i, r_i)|\mathbf{v}_{<i}^{\mathcal{R}})$ and treat each $K \times M$ possible visual word/region pair as a distinct word. One implication of this is that the binary tree of visual words must be larger so as to have a leaf for each possible visual word/region pair. Fortunately, since computations grow logarithmically with the size of the tree, this is not a problem and we can still deal with a large number of regions.

4.3. Dealing with Annotations

So far, we've described how to model the visual word and class label modalities. In this section, we now describe how we also model the annotation word modality with SupDocNADE.

Specifically, let $\mathcal{A}$ be the predefined vocabulary of all annotation words, we will note the annotation of a given image as $\mathbf{a} = [a_1, a_2, \ldots, a_L]$ where $a_i \in \mathcal{A}$, with L being the number of words in the annotation. Thus, the image with its annotation can be represented as a mixed bag of visual and annotation words:

$$\begin{aligned} \mathbf{v}^{\mathcal{A}} &= [v_1^{\mathcal{A}}, \ldots, v_D^{\mathcal{A}}, v_{D+1}^{\mathcal{A}}, \ldots, , v_{D+L}^{\mathcal{A}}] \\ &= [v_1^{\mathcal{R}}, \ldots, v_D^{\mathcal{R}}, a_1, \ldots, a_L] \ . \end{aligned} \qquad (16)$$

To embed the annotation words into the SupDocNADE framework, we treat each annotation word the same way we deal with visual words. Specifically, we use a joint indexing of all visual and annotation words and use a larger binary word tree so as to augment it with leaves for the annotation words. By training SupDocNADE on this joint image/annotation representation $\mathbf{v}^{\mathcal{A}}$, it can learn the relationship between the labels, the spatially-embedded visual words and the annotation words.

At test time, the annotation words are not given and we wish to predict them. To achieve this, we compute the document representation $\mathbf{h}_y(\mathbf{v}^{\mathcal{R}})$ based only on the visual words and compute for each possible annotation word $a \in \mathcal{A}$ the probability that it would be the next observed word $p(v_i^{\mathcal{A}} = a|\mathbf{v}^{\mathcal{A}} = \mathbf{v}^{\mathcal{R}})$, based on the tree decomposition as in Equation 4. In other words, we only compute the probability of paths that reach a leaf corresponding to an annotation word (not a visual word). We then rank the annotation words in $\mathcal{A}$ in decreasing order of their probability and select the top 5 words as our predicted annotation.

5. Experiments and Results

To test the ability of SupDocNADE to learn from multimodal data, we measured its performance under simultaneous image classification and annotation tasks. We tested our model on 2 real-world datasets: a subset of the LabelMe dataset [21] and the UIUC-Sports dataset [22]. LabelMe and UIUC-Sports come with annotations and are popular classification and annotation benchmarks. We performed extensive quantitative comparisons of SupDocNADE with the original DocNADE model and supervised LDA (sLDA)[1] [10, 9]. We also provide some comparisons with MMLDA [12] and a Spatial Pyramid Matching (SPM) approach [17]. The code to download the datasets and for SupDocNADE is available at https://sites.google.com/site/zhengyin1126/.

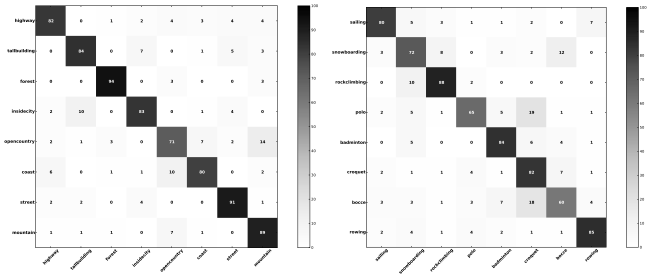
5.1. Datasets Description

Following Wang et al. [9], we constructed our LabelMe dataset using the online tool to obtain images of size 256 × 256 pixels from the following 8 classes: *highway*, *inside city*, *coast*, *forest*, *tall building*, *street*, *open country* and *mountain*. For each class, 200 images were randomly selected and split evenly in the training and test sets, yielding a total of 1600 images.

The UIUC-Sports dataset contains 1792 images, classified into 8 classes: *badminton* (313 images), *bocce* (137 images), *croquet* (330 images), *polo* (183 images), *rockclimbing* (194 images), *rowing* (255 images), *sailing* (190 images), *snowboarding* (190 images). Following previous work, the maximum side of each image was resized to 400 pixels, while maintaining the aspect ratio. We randomly split the images of each class evenly into training and test sets. For both LabelMe and UIUC-Sports datasets, we removed the annotation words occurring less than 3 times, as in Wang et al. [9].

5.2. Experimental Setup

Following Wang et al. [9], 128 dimensional, densely extracted SIFT features were used to extract the visual words. The step and patch size of the dense SIFT extraction was set to 8 and 16, respectively. The dense SIFT features from the training set were quantized into 240 clusters, to construct our visual word vocabulary, using K-means. We divided each image into a 2×2 grid to extract the spatial position information, as described in Section 4.2. This produced $2 \times 2 \times 240 = 960$ different visual word/region pairs.



	highway	tallbuilding	forest	insidecity	opencountry	coast	street	mountain
highway	82	0	1	2	4	3	4	4
tallbuilding	0	84	0	7	0	1	5	3
forest	0	0	94	0	3	0	0	3
insidecity	2	10	0	83	0	1	4	0
opencountry	2	1	3	0	71	7	2	14
coast	6	0	1	1	10	80	0	2
street	2	2	0	4	0	0	91	1
mountain	1	1	1	0	7	1	0	89

	sailing	snowboarding	rockclimbing	polo	badminton	croquet	bocce	rowing
sailing	80	5	3	1	1	2	0	7
snowboarding	3	72	8	0	3	2	12	0
rockclimbing	0	10	88	2	0	0	0	0
polo	2	5	1	65	5	19	1	1
badminton	0	5	0	0	84	6	4	1
croquet	2	1	1	4	1	82	7	1
bocce	3	3	1	3	7	18	60	4
rowing	2	4	1	4	2	1	1	85

Figure 2. The confusion matrix for LabelMe (left) and UIUC-Sports (right) datasets.

We use classification accuracy to evaluate the performance of image classification and the average F-measure of the top 5 predicted annotations to evaluate the annotation performance, as in previous work. The F-measure of an image is defined as

$$F\text{-measure} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{17}$$

where recall is the percentage of correctly predicted annotations out of all ground-truth annotations for an image, while the precision is the percentage of correctly predicted annotations out of all predicted annotations[2]. We used 5 random train/test splits to estimate the average accuracy and F-measure.

Image classification with SupDocNADE is performed by feeding the learned document representations to a RBF kernel SVM. In our experiments, all hyper-parameters (learning rate, unsupervised learning weight λ in SupDocNADE, C and γ in RBF kernel SVM), were chosen by cross validation. We emphasize that, again from following Wang et al. [9], the annotation words are not available at test time and all methods predict an image's class based solely on its bag of visual words, .

5.3. Quantitative Comparison

In this section, we describe our quantitative comparison between SupDocNADE, DocNADE and sLDA. We used the implementation of sLDA available at http://www.cs.cmu.edu/~chongw/slda/ in our comparison, to which we fed the same visual (with spatial regions) and annotation words as for DocNADE and SupDocNADE.

The classification results are illustrated in Figure 3. Similarly, we observe that SupDocNADE outperforms DocNADE and sLDA. Tuning the trade-off between generative

[1] We mention that [9] has shown that sLDA performs better than Corr-LDA[2]. Moreover, [4] found that Multimodal LDA [5] did not improve on the performance of Corr-LDA. Finally, sLDA distinguishes itself from the other models in the fact that it also supports the class label modality and has code available online. Hence, we compare directly with sLDA only.

[2] When there are repeated words in the ground-truth annotations, the repeated terms were removed to calculate the F-measure.

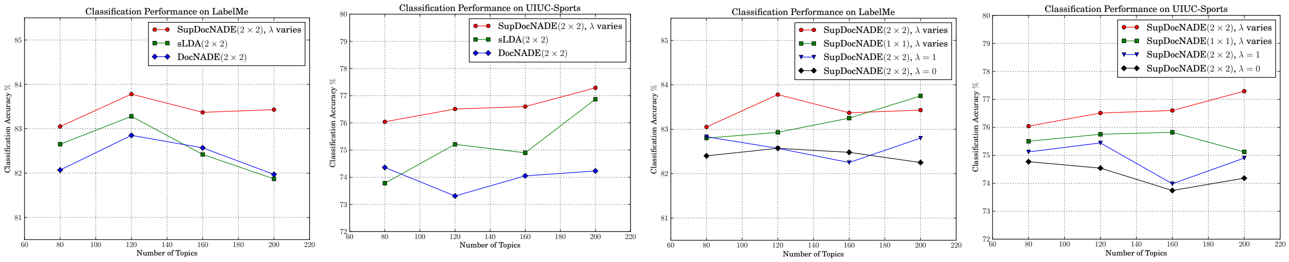


Figure 3. Classification performance comparison on LabelMe (even) and UIUC-Sports (odd). On the left, we compare the classification performance of SupDocNADE, DocNADE and sLDA. On the right, we compare the performance between different variants of SupDocNADE. The "λ *varies*" means the unsupervised weight λ in Equation 12 is chosen by cross-validation.

Table 1. Performance comparison of the different models.

Model	LabelMe		UIUC-Sports	
	Accuracy	F-measure	Accuracy	F-measure
SPM [17]	80.88%	43.68%	72.33%	41.78%
MMLDA [12]	$81.47\%^{\dagger}$	$\mathbf{46.64\%}^{\dagger *}$	$74.65\%^{\dagger}$	$44.51\%^{\dagger}$
sLDA [9]	81.87%	$38.7\%^{\dagger}$	76.87%	$35.0\%^{\dagger}$
DocNADE	81.97%	43.32%	74.23%	46.38%
SupDocNADE	$\mathbf{83.43\%}$	43.87%	$\mathbf{77.29\%}$	$\mathbf{46.95\%}$

$\dagger$: Taken from the original paper.

$*$: MMLDA performs classification and annotation separately and doesn't learn jointly from all 3 modalities.

and discriminative learning and exploiting position information is usually beneficial. There is just one exception, on LabelMe, with 200 hidden topic units, where using a 1×1 grid slightly outperforms a 2×2 grid.

As for image annotation, we computed the performance of our model with 200 topics. As shown in Table 1, SupDocNADE obtains an F-measure of 43.87% and 46.95% on the LabelMe and UIUC-Sports datasets respectively. This is slightly superior to regular DocNADE. Since code for performing image annotation using sLDA is not publicly available, we compare directly with the results found in the corresponding paper [9]. Wang et al. [9] report F-measures of 38.7% and 35.0% for sLDA, which is below SupDocNADE by a large margin.

We also compare with MMLDA [12], which has been applied to image classification and annotation separately. The reported classification accuracy for MMLDA is less than SupDocNADE as shown in Table 1. The performance for annotation reported in [12] is better than SupDocNADE on LabelMe but worse on UIUC-Sports. We highlight that MMLDA did not deal with the class label and annotation word modalities *jointly*, the different modalities being treated separately.

The spatial pyramid approach of [17] could also be adapted to perform both image classification and annotation. We used the code from [17] to generate 2 layer-SPM representations with a vocabulary size of 240, which is the same configuration as used by the other models. For image classification, an SVM with Histogram Intersection Kernel (HIK) is adopted as the classifier, as in Lazebnik et al. [17]. For annotation, we used a k nearest neighbor (KNN) prediction of the annotation words for the test images. Specifically, the top 5 most frequent annotation words among the k nearest images (based on the SPM representation with HIK similarity) in the training set were selected as the prediction of a test image's annotation words. The number k was selected by cross validation, for each of the 5 random splits. As shown in Table 1, SPM achieves a classification accuracy of 80.88% and 72.33% for LabelMe and UIUC-Sports, which is lower than SupDocNADE. As for annotation, the F-measure of SPM is also lower than SupDocNADE, with 43.68% and 41.78% for LabelMe and UIUC-Sports, respectively.

Figure 4 illustrates examples of correct and incorrect predictions made by SupDocNADE on the LabelMe dataset. Figure 2 also provides the classification confusion matrix on both LabelMe and UIUC-Sports benchmarks.

5.4. Visualization of Learned Representations

Since topic models are often used to interpret and explore the semantic structure of image data, we looked at how we could observe the structure learned by SupDocNADE.

We extracted the visual/annotation words that were most strongly associated with certain class labels within SupDocNADE as follows. Given a class label *street*, which corresponds to a column $\mathbf{U}_{:,i}$ in matrix $\mathbf{U}$, we selected the top 3 topics (hidden units) having the largest connection weight in $\mathbf{U}_{:,i}$. Then, we averaged the columns of matrix $\mathbf{W}$ corresponding to these 3 hidden topics and selected the visual/annotation words with largest averaged weight connection. The results of this procedure for classes *street*, *sailing*, *forest* and *highway* is illustrated in Figure 5. To visualize the visual words, we show 16 image patches belonging to each visual word's cluster, as extracted by K-means. The

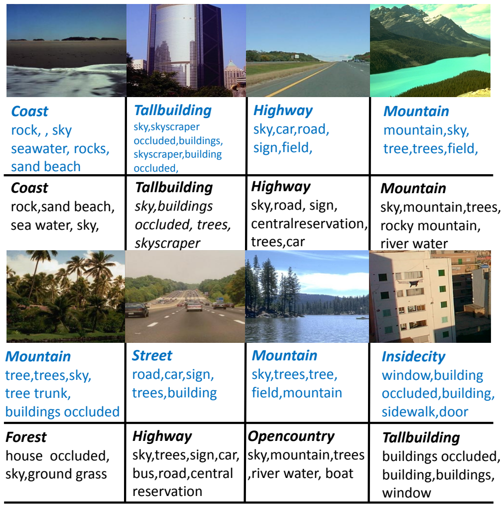


Figure 4. Predicted class and annotation by SupDocNADE on LabelMe dataset. We list some correctly (top row) and incorrectly (bottom row) classified images. The predicted (in blue) and ground-truth (in black) class labels and annotation words are presented under each image.

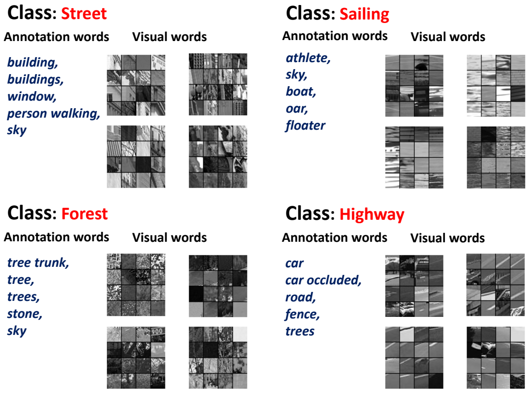


Figure 5. Visualization of learned representations. Class labels are colored in red. For each class, we list 4 visual words (each represented by 16 image patches) and 5 annotation words that are strongly associated with each class. See Sec. 5.4 for more details.

learned associations are intuitive: for example, the class *street* is associated with the annotation words "*building*", "*buildings*", "*window*", "*person walking*" and "*sky*", while the visual words showcase parts of buildings and windows.

6. Conclusion and Discussion

In this paper, we proposed SupDocNADE, a supervised extension of DocNADE, which can learn jointly from visual words, annotations and class label. Like all topic models, our model is trained to model the distribution of the bag of words representation of images and can extract a meaningful representation from it. Unlike most topic models however, the image representation is not modeled as a latent random variable in a model, but instead as the hidden layer of a neural autoregressive network. A distinctive advantage of SupDocNADE is that it does not require any iterative, approximate inference procedure to compute an image's representation. Our experiments confirm that SupDocNADE is a competitive approach for multimodal data modeling.

References

[1] K. Barnard, P. Duygulu, D. Forsyth, *et al.*, "Matching words and pictures," *JMLR*, 2003. 1

[2] D. M. Blei and M. I. Jordan, "Modeling annotated data," in *ACM SIGIR*, 2003. 1, 2, 6

[3] R. Socher and L. Fei-Fei, "Connecting modalities: Semi-supervised segmentation and annotation of images using unaligned text corpora," in *CVPR*, 2010. 1

[4] Y. Jia *et al.*, "Learning cross-modality similarity for multinomial data," in *ICCV*, 2011. 1, 2, 6

[5] D. Putthividhy *et al.*, "Topic regression multi-modal latent dirichlet allocation for image annotation," in *CVPR*, 2010. 1, 2, 6

[6] M. Guillaumin *et al.*, "Multimodal semi-supervised learning for image classification," in *CVPR*, 2010. 1

[7] N. Rasiwasia *et al.*, "A new approach to cross-modal multimedia retrieval," in *ACM-MM*, 2010. 1

[8] D. Blei *et al.*, "Latent dirichlet allocation," *JMLR*, 2003. 1

[9] C. Wang, D. Blei, and F.-F. Li, "Simultaneous image classification and annotation," in *CVPR*, 2009. 1, 2, 6, 7

[10] D. M. Blei and J. D. McAuliffe, "Supervised topic models," in *NIPS*, 2007. 2, 4, 6

[11] H. Larochelle and S. Lauly, "A neural autoregressive topic model," in *NIPS 25*, 2012. 2, 3, 4

[12] Y. Wang *et al.*, "Max-margin latent dirichlet allocation for image classification and annotation," in *BMVC*, 2011. 2, 6, 7

[13] Y. Bengio *et al.*, "Representation learning: A review and new perspectives," *arXiv preprint arXiv:1206.5538*, 2012. 2

[14] R. Salakhutdinov and G. E. Hinton, "Replicated softmax: an undirected topic model," in *NIPS*, 2009. 2

[15] N. Srivastava and R. Salakhutdinov, "Multimodal learning with deep boltzmann machines." in *NIPS*, 2012. 2

[16] J. Ngiam *et al.*, "Multimodal deep learning," in *ICML*, 2011. 2

[17] S. Lazebnik *et al.*, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *CVPR*, 2006. 4, 5, 6, 7

[18] G. Bouchard and B. Triggs, "The tradeoff between generative and discriminative classifiers," in *COMPSTAT*, 2004. 4

[19] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier networks," in *AISTATS*, 2011. 4

[20] V. Nair and G. E. Hinton, "Rectified linear units improve restricted boltzmann machines," in *ICML*, 2010. 4

[21] B. C. Russell *et al.*, "Labelme: a database and web-based tool for image annotation," *IJCV*, 2008. 6

[22] L.-J. Li and L. Fei-Fei, "What, where and who? classifying events by scene and object recognition," in *ICCV*, 2007. 6