Tree-Structured Topic Modeling with Nonparametric Neural Variational Inference

Ziye Chen[1], **Cheng Ding**[1], **Zusheng Zhang**[1], **Yanghui Rao**[1,*], **Haoran Xie**[2]
[1]School of Computer Science and Engineering, Sun Yat-sen University, Guangzhou, China
[2]Department of Computing and Decision Sciences, Lingnan University, Hong Kong
{chenzy35,dingch6,zhangzsh3}@mail2.sysu.edu.cn,
raoyangh@mail.sysu.edu.cn, hrxie2@gmail.com

Abstract

Topic modeling has been widely used for discovering the latent semantic structure of documents, but most existing methods learn topics with a flat structure. Although probabilistic models can generate topic hierarchies by introducing nonparametric priors like Chinese restaurant process, such methods have data scalability issues. In this study, we develop a tree-structured topic model by leveraging nonparametric neural variational inference. Particularly, the latent components of the stick-breaking process are first learned for each document, then the affiliations of latent components are modeled by the dependency matrices between network layers. Utilizing this network structure, we can efficiently extract a tree-structured topic hierarchy with reasonable structure, low redundancy, and adaptable widths. Experiments on real-world datasets validate the effectiveness of our method.

1 Introduction

Topic models (Blei et al., 2003; Griffiths et al., 2004) are important tools for discovering latent semantic patterns in a corpus. These models can be grouped into flat models and hierarchical models. In many domains, topics can be naturally organized into a tree, where the hierarchical relationships among topics are valuable for data analysis and exploration. Tree-structured topic model (Griffiths et al., 2004) was thus developed to learn coherent topics from text without disrupting the inherent hierarchical structure. Such a method has been proven as useful in various downstream applications, including hierarchical categorization of Web pages (Ming et al., 2010), aspects hierarchies extraction in reviews (Kim et al., 2013), and hierarchies discovery of research topics in academic repositories (Paisley et al., 2014).

Despite the practical importance and potential advantages, tree-structured topic models still face the following challenges. Firstly, the hierarchical structure of topics should be reasonable (Viegas et al., 2020). Typically, topics near the root are more general while the ones close to the leaves are more specific. Besides, child topics should be coherent with their corresponding parent topics. Secondly, low redundancy is necessary for the extracted topics, in order to prevent the distributions associated with parent topics and their children being extremely similar (Griffiths et al., 2004). Thirdly, the number of topics in each hierarchy level should be automatically determined by the model, because it is usually unknown and can not be previously set to a predefined value (Kim et al., 2012). Finally, it is difficult for probabilistic models to enhance the data scalability (Isonuma et al., 2020). Previously, several tree-structured topic models (Griffiths et al., 2004; Kim et al., 2012; Isonuma et al., 2020) have been developed. But these methods can not fully overcome the aforementioned challenges.

In this paper, we focus on grouping topics into a reasonable tree structure, based on the neural variational inference (NVI) framework (Kingma and Welling, 2014; Rezende et al., 2014) with a nonparametric prior. Owing to the excellent function fitting ability, neural network has been widely introduced into topic modeling. Nonetheless, few neural methods explicitly model the dependencies among different layers and get explainable hierarchical topics, which is largely due to the weak interpretability of neural networks. Furthermore, the inflexibility of neural networks also makes it difficult to learn an unbounded number of topics at each level. To address these limitations, we propose a novel nonparametric neural method to generate tree-structured topic hierarchies, namely nonparametric Tree-Structured Neural Topic Model

*The corresponding author.

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, pages 2343–2353
August 1–6, 2021.

(nTSNTM)[1]. By connecting the network layers with dependency matrices, the model is able to extract an explainable tree-structured hierarchy. Firstly, the topic affiliations among hierarchy levels can be determined by the dicrete vectors of the dependency matrices. Secondly, to control redundancy among topics, we allow the model to freely generate topics without duplicating their corresponding parent topics. Thirdly, we couple a stick-breaking process with NVI to equip the topic tree with self-determined widths, which can help the model determine the number of topics automatically. Finally, due to the advantages of neural networks, our model can scale to larger datasets conveniently. Experiments indicate that our model outperforms baselines on several widely adopted metrics and two new measurements developed for tree-structured topic models.

The rest of this paper is organized as follows. We describe related work in Section 2. Then, we detail the proposed nTSNTM in Section 3. Section 4 presents our experimental results and discussions. Finally, we draw conclusions in Section 5.

2 Related Work

In (Griffiths et al., 2004), a tree-structured topic model called hLDA was first proposed by introducing a nested Chinese restaurant process (nCRP). For hLDA, a topic tree is constructed through Gibbs sampling given a certain depth. Based on hLDA, Xu et al. (2018) proposed a knowledged-based HTM to generate topic hierarchies from multiple domains corpora, but the hierarchical relation between the ancestor topic and the offspring one may be unclear, because a document is generated by the topics along a single path of the tree. To overcome this issue, Kim et al. (2012) proposed a recursive CRP (rCRP), in which a document possesses a distribution over the entire tree. Although rCRP has shown remarkable competitiveness in hierarchical topic modeling, it suffers from the major limitation of data scalability (Isonuma et al., 2020). Several other methods focused on hierarchical text clustering. For instance, Ghahramani et al. (2010) applied nested stick-breaking processes to cluster data into a tree structure. Unfortunately, the above method only models a document by a single node of the tree. Liu et al. (2014) developed a model named HLTA for topic detection, in which words and topics are clustered by employing the Bridged-Islands algorithm iteratively. However, HLTA is unable to cope with polysemous words, which is quite important for topic models.

To couple nonparametric processes with NVI, Miao et al. (2017) used Gaussian distributions to generate stick-breaking fractions. Nalisnick and Smyth (2017) first described how to use stochastic gradient variational Bayes for posterior inference of the weights in stick-breaking processes. Experiments indicated that the latent representations of the above model were more discriminative than those of the Gaussian variant. Then, Ning et al. (2020) developed two nonparametric neural topic models by treating topics as trainable parameters. Unfortunately, the aforementioned methods can only learn topics with a flat structure.

For tree-structured neural topic modeling, a feasible way is to decompose the distribution over the topic tree into a path distribution and a level distribution. Following (Wang and Blei, 2009), where a tree-based stick-breaking construction of nCRP was first derived to draw topic paths, and then a level distribution was learned to sample topics along the path, Isonuma et al. (2020) proposed a tree-structured neural topic model (TSNTM) by parameterizing an unbounded ancestral and fraternal topic distribution. TSNTM applies a doubly-recurrent neural network (DRNN) to obtain topic embeddings via ancestral and fraternal edges, then generates breaking fractions by the dot product between document embeddings and topic embeddings. However, TSNTM fails to learn a reasonable topic tree for the following reasons. Firstly, the breaking fractions do not obey the Beta distributions adopted in the stick-breaking process (SBP). Secondly, the structure of DRNN in TSNTM is simplified, where the topic embeddings are generated directly by an initialized root embedding and two parameter matrices (i.e., ancestral and fraternal connections). This prevents the model from learning appropriate semantic embeddings for topics. Finally, TSNTM relies on heuristic rules to update the tree structure.

Another stream of work is to generate a document by a directed acyclic graph (DAG) structured topic hierarchy. For instance, Li and McCallum (2006) introduced the pachinko allocation model (PAM) to capture correlations between topics using a DAG. Mimno et al. (2007) proposed the hierarchical PAM by connecting the root topic to lower-level

[1]The code of our model is available in public at: https://github.com/hostnlp/nTSNTM.

topics through multinomial distributions. Nonprobabilistic matrix factorization was also used to extract the topic structure. Liu et al. (2018) used non-negative matrix factorization (NMF) with three optimization constraints, including global independence, local independence, and information consistency, to preserve topic coherence and a reasonable structure. Viegas et al. (2020) incorporated pre-trained word embeddings into NMF to further improve topic coherence. The main limitation of NMF-based methods, however, is that a time-consuming process (e.g., measure the stability of results by running multiple random samplings) is necessary to determine the number of topics at each level. This is because nonparametric priors are intractable to be included in these models.

3 Tree-Structured Neural Topic Model with Nonparametric Prior

In this section, we firstly describe the stick-breaking process. Then, we introduce the modeling of tree-structured topic hierarchy. Finally, we detail the inference method of our nTSNTM.

3.1 Stick-breaking Process

For nonparametric models, stick-breaking prior is a random measure with the form $G = \sum_{k=1}^{\infty} \pi_k \delta_{\zeta_k}$, where δ_{ζ_k} is a discrete measure concentrated at $\zeta_k \sim G_0$ (Ishwaran and James, 2001)[2], i.e, a draw from the base measure. The π_ks are random weights independent of G_0 (Nalisnick and Smyth, 2017). This constructive definition is known as SBP (Sethuraman, 1994), which implies that the weights $\boldsymbol{\pi} = (\pi_k)_{k=1}^{\infty}$ can be drawn according to the procedure of iteratively breaking off segments from a unit stick.

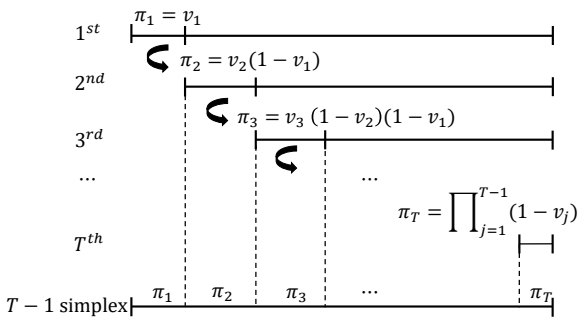

Figure 1: Stick-breaking construction.

[2]In topic models, ζ_k represents the k^{th} topic and G_0 represents the topic space.

As shown in Figure 1, we break the unit stick and get the first component with length v_1. If a fraction v_2 of the remaining stick is broken off, then we obtain the second component with length $v_2(1 - v_1)$ and a remaining stick with length $(1 - v_1)(1 - v_2)$. The following breaks are taken on the remaining stick by the same operation. Given a truncation level T, the length of the last component will be $\prod_{j=1}^{T-1}(1 - v_j)$. Formally, the length of each component is defined as:

$$\pi_k = \begin{cases} v_1 & \text{if} \quad k = 1, \\ v_k \prod_{t<k}(1 - v_t) & \text{for} \quad k > 1, \end{cases} \tag{1}$$

where $v_k \sim \text{Beta}(\boldsymbol{\alpha}_0, \boldsymbol{\beta}_0)$, with $\boldsymbol{\alpha}_0$ and $\boldsymbol{\beta}_0$ being the prior parameters. Note that the component weights $\boldsymbol{\pi}$ satisfy $0 \leq \pi_k \leq 1$ and $\sum_{k=1}^{\infty} \pi_k = 1$, thus we can interpret $\boldsymbol{\pi}$ as random probabilities. Particularly, when $v_k \sim \text{Beta}(1, \boldsymbol{\beta}_0)$, the joint distribution for $\boldsymbol{\pi}$ is the GEM distribution (Pitman, 2006) with concentration parameter $\boldsymbol{\beta}_0$, and the corresponding SBP is one of the constructions for the Dirichlet process, a popular nonparametric random process for topic modeling (Teh et al., 2005).

In our method, we take component weights $\boldsymbol{\pi}$ as the path distribution of a document. We assume that the words of a document come from several topic paths. Due to the sequentiality of the stick-breaking operation, paths with smaller serial numbers are more likely to be activated to represent the documents, while paths with larger serial numbers tend to be unactivated. The number of activated paths can be adjusted by SBP automatically.

3.2 Tree-Structured Topic Hierarchy

To conveniently describe our method, we here compare the sampling processes for an example document of different tree-structured topic models. As shown in Figure 2, hLDA (Griffiths et al., 2004) considers that a document is generated by topics of a single path, which violates the multi-topics assumption of topic models (i.e., a document may span several topics). Considering this issue, rCRP (Kim et al., 2012) and TSNTM (Isonuma et al., 2020) assume that a document can be generated by any topic in the tree. We follow the above assumption adopted in rCRP and TSNTM to model a tree-structured topic hierarchy, but the difference is that our model takes the sampling from the bottom up rather than from the top down as in rCRP and TSNTM. Particularly, rCRP samples topics from the root using recursive CRP. TSNTM samples

paths from the root by applying a DRNN (Alvarez-Melis and Jaakkola, 2017), and it needs to update the tree structure frequently by heuristic rules. On the contrary, our model directly samples the leaf topics, and the paths toward the root are determined automatically. Specifically, we use a common stick-breaking construction to infer the distribution over leaf topics, which corresponds to the path distribution. Besides, we use dependency matrices to keep track of the affiliations among topics. Thus the tree structure can be updated through back propagation.

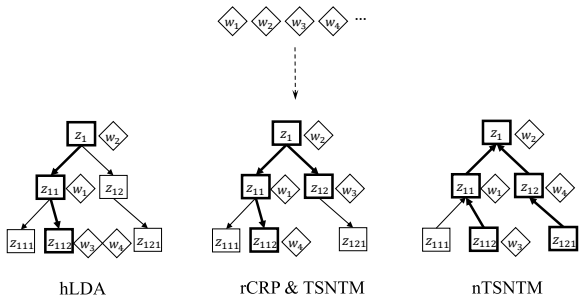

Figure 2: Sampling process of an example document for hLDA (Griffiths et al., 2004), rCRP (Kim et al., 2012), TSNTM (Isonuma et al., 2020), and our nTSNTM. Each node represents a topic z with its distributions over words $\boldsymbol{w}$. The active topics and path are highlighted by boldface.

Figure 3 shows the graphical representation of nTSNTM. For our model, the number of leaf topics is determined by SBP, and the numbers of non-leaf topics are adjusted through dependency matrices $\mathbf{M}$ between network layers. The l^{th} item of $\mathbf{M}$, i.e., $\mathbf{M}_l \in [0,1]^{K_l * K_{l+1}}$, is the dependency matrix between layers l and l+1, where K_l and K_{l+1} represent the maximum numbers of topics at level l and level l+1, respectively. In particular, $M_{l,k,j}$ is the probability of topic j at level l being the parent of topic k at level l+1 with $\sum_{j'} M_{l,k,j'} = 1$. As mentioned in (Griffiths et al., 2004), a clear tree structure indicates that each sub-topic has a relationship with no more than one super-topic. So a softmax function with low temperature (Hinton et al., 2015) is applied to ensure that $\boldsymbol{M}_{l,k}$ approximates a discrete one-hot vector. In this way, the topic tree can be built through the introduced $\mathbf{M}$ from bottom up. Furthermore, the topic hierarchy can be updated automatically according to the update of $\mathbf{M}$.

After determining the topic hierarchy by $\mathbf{M}$, the generative process of each word in nTSNTM can be described as follows:

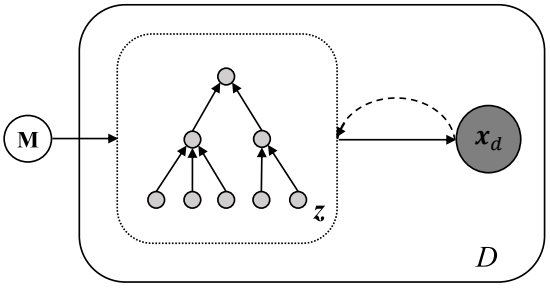

Figure 3: Graphical representation of nTSNTM. Solid and dashed arrows denote generation and inference.

1. For each document $\boldsymbol{x}_d \in \{\boldsymbol{x}_1, ..., \boldsymbol{x}_D\}$:

$$\text{Draw SBP weights: } \boldsymbol{\pi}_d \sim \text{GEM}(\boldsymbol{\beta}_0); \quad (2)$$
$$\text{Draw Gaussian samples: } \boldsymbol{g}_d \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}^2); \quad (3)$$
$$\text{Draw level distributions: } \boldsymbol{\eta}_d = f_\eta(\boldsymbol{g}_d). \quad (4)$$

2. For each word $w_{d,n} \in \{w_{d,1}, ..., w_{d,N_d}\}$ in $\boldsymbol{x}_d$:

$$\text{Draw a path: } c_{d,n} \sim \text{Multi}(\boldsymbol{\pi}_d); \quad (5)$$
$$\text{Draw a level: } r_{d,n} \sim \text{Multi}(\boldsymbol{\eta}_d); \quad (6)$$
$$\text{Draw a word: } w_{d,n} \sim \text{Multi}(\boldsymbol{\phi}_{c_{d,n}[r_{d,n}]}). \quad (7)$$

In the above, D is the number of documents, N_d is the number of words in $\boldsymbol{x}_d$. $\boldsymbol{\phi}_{c_{d,n}[r_{d,n}]} \in \triangle^{V-1}$ is the word distribution of the topic at level $r_{d,n}$ of path $c_{d,n}$, and V is the vocabulary size. $f_\eta(\cdot)$ is a neural perceptron with softmax activation to transform a Gaussian sample to a level distribution.

3.3 Parameter Inference

Since the Beta distribution does not have a differentiable non-centered parametrization that NVI requires (Kingma and Welling, 2014), we choose the Kumaraswamy distribution (Kumaraswamy, 1980) to approximate $\text{GEM}(\boldsymbol{\beta}_0)$, i.e., the conjunction of $\text{Beta}(1, \boldsymbol{\beta}_0)$ and a stick-breaking operation (Nalisnick and Smyth, 2017). For the Kumaraswamy distribution, the probability density function on the unit interval is defined as $\text{Kumaraswamy}(x; a, b) = abx^{a-1}(1-x^a)^{b-1}$ for $x \in (0,1)$ and $a, b > 0$. Samples can be drawn via the inverse transform: $x \sim (1-u^{\frac{1}{b}})^{\frac{1}{a}}$ where $u \sim \text{Uniform}(0,1)$. Then the KL-divergence between the Kumaraswamy distribution and the Beta distribution can be closely approximated in the closed-form. We describe the parameter inference process of our nTSNTM as follows.

Firstly, we estimate the component weights of document $\boldsymbol{x}_d$, i.e., $\hat{\boldsymbol{\pi}}_d$, by the following stick-

breaking operation with fractions $\boldsymbol{v}_d$:

$$\boldsymbol{\alpha}_d = f_\alpha(\boldsymbol{x}_d), \quad \boldsymbol{\beta}_d = f_\beta(\boldsymbol{x}_d), \tag{8}$$

$$\boldsymbol{v}_d \sim \text{Kumaraswamy}(\boldsymbol{\alpha}_d, \boldsymbol{\beta}_d), \tag{9}$$

$$\hat{\boldsymbol{\pi}}_d = (v_{d,1}, ..., \prod_{j=1}^{T-1}(1 - v_{d,j})), \tag{10}$$

where the bag-of-words representation is used for $\boldsymbol{x}_d$. To ensure positive outputs, $f_\alpha(\cdot)$ and $f_\beta(\cdot)$ are neural perceptrons with softplus activation.

Secondly, we infer the level distributions $\hat{\boldsymbol{\eta}}_d$ by:

$$\boldsymbol{\mu}_d = f_\mu(\boldsymbol{x}_d), \quad \boldsymbol{\sigma}_d = f_\sigma(\boldsymbol{x}_d), \tag{11}$$

$$\hat{\boldsymbol{g}}_d \sim \mathcal{N}(\boldsymbol{\mu}_d, \boldsymbol{\sigma}_d^2), \quad \hat{\boldsymbol{\eta}}_d = f_\eta(\hat{\boldsymbol{g}}_d), \tag{12}$$

where $f_\mu(\cdot)$ and $f_\sigma(\cdot)$ are linear transformations. In practice, we reparameterize $\hat{\boldsymbol{g}}_d = \boldsymbol{\mu}_d + \hat{\epsilon} * \boldsymbol{\sigma}_d$ with the sample $\hat{\boldsymbol{\epsilon}} \sim \mathcal{N}(\mathbf{0}, \boldsymbol{I}^2)$ (Rezende et al., 2014).

Thirdly, we obtain the topic distributions of $\boldsymbol{x}_d$, i.e., $\hat{\boldsymbol{\theta}}_d = \{\hat{\boldsymbol{\theta}}_{d,1}, ..., \hat{\boldsymbol{\theta}}_{d,L}\}$ by:

$$\hat{\boldsymbol{\theta}}_{d,l} = \begin{cases} \hat{\eta}_{d,L}\hat{\boldsymbol{\pi}}_d & \text{if} \quad l = L, \\ \hat{\eta}_{d,l}\hat{\boldsymbol{\pi}}_d \prod_{l' \geq l} \mathbf{M}_{l'} & \text{for} \quad l < L, \end{cases} \tag{13}$$

where L denotes the depth of the topic tree, and $\sum_l \sum_k \hat{\theta}_{d,l,k} = 1$.

Then, we follow (Miao et al., 2017) to explicitly model topic-word distributions by: $\boldsymbol{\phi} = softmax(\boldsymbol{u} * \boldsymbol{t}^T)$, where $\boldsymbol{u} \in \mathbb{R}^{V*H}$ and $\boldsymbol{t} \in \mathbb{R}^{\sum_l K_l * H}$ are word vectors and topic vectors, and H denotes the dimension of word/topic vectors. Given topic-word distributions $\boldsymbol{\phi}$ and topic distributions $\hat{\boldsymbol{\theta}}_d$ obtained from Eq. (13), our model reconstructs each document $\boldsymbol{x}_d$ by: $p(w_{d,n}|\boldsymbol{\phi}, \hat{\boldsymbol{\theta}}_d) = \sum_{z_{d,n}} [p(w_{d,n}|\boldsymbol{\phi}_{z_{d,n}}) p(z_{d,n}|\hat{\boldsymbol{\theta}})] = \hat{\boldsymbol{\theta}}_d * \boldsymbol{\phi}$, where $z_{d,n}$ is the topic assignment for $w_{d,n}$.

Finally, the variational lower-bound of $\boldsymbol{x}_d$ is:

$$\begin{aligned} \mathcal{L} =& \mathbb{E}_{q(\boldsymbol{\pi}_d, \boldsymbol{\eta}_d|\boldsymbol{x}_d)}[\sum_n \log(p(w_{d,n}|\boldsymbol{\phi}, \hat{\boldsymbol{\theta}}_d))] \\ &- D_{KL}[q(\boldsymbol{\pi}_d|\boldsymbol{x}_d)||p(\boldsymbol{\pi}_d)] \\ &- D_{KL}[q(\boldsymbol{\eta}_d|\boldsymbol{x}_d)||p(\boldsymbol{\eta}_d)], \end{aligned} \tag{14}$$

where $q(\boldsymbol{\pi}_d|\boldsymbol{x}_d)$ and $q(\boldsymbol{\eta}_d|\boldsymbol{x}_d)$ are posteriors modeled by the inference network. $p(\boldsymbol{\pi}_d)$ is the prior for $\boldsymbol{\pi}_d$, i.e., $\text{GEM}(\boldsymbol{\beta}_0)$, and $p(\boldsymbol{\eta})$ is the prior for $\boldsymbol{\eta}$, i.e., the standard Gaussian transformed by $f_\eta(\cdot)$.

The parameter inference method for nTSNTM is presented in Algorithm 1. We use the variational lower-bound to calculate gradients and apply Adam (Kingma and Ba, 2015) to update parameters.

Algorithm 1: Parameter Inference Algorithm

Input: GEM priors $\boldsymbol{\beta}_0$ and documents $\{\boldsymbol{x}_1, ..., \boldsymbol{x}_D\}$;
Output: Document-topic distribution $\boldsymbol{\theta}$, topic-word distribution $\boldsymbol{\phi}$, and topic tree $\boldsymbol{Tr}$.

Randomly initialize dependency matrices $\mathbf{M}$ and topic-word distribution $\boldsymbol{\phi}$;
repeat
  for *document* $\boldsymbol{x}_d \in \{\boldsymbol{x}_1, ..., \boldsymbol{x}_D\}$ **do**
    Estimate $\hat{\boldsymbol{\pi}}_d$ and $\hat{\boldsymbol{\eta}}_d$ by Eqs. (8–12);
    Compute $\hat{\boldsymbol{\theta}}_d$ by Eq. (13);
    for $w_{d,n} \in \boldsymbol{x}_d$ **do**
      $p(w_{d,n}|\hat{\boldsymbol{\theta}}_{\boldsymbol{d}}, \boldsymbol{\phi}) = \hat{\boldsymbol{\theta}}_d * \boldsymbol{\phi}$;
    end
    Compute $\mathcal{L}$ by Eq. (14);
    Update $f_\alpha(\cdot)$, $f_\beta(\cdot)$, $f_\mu(\cdot)$, $f_\sigma(\cdot)$, $f_\eta(\cdot)$, $\boldsymbol{\phi}$, and $\mathbf{M}$;
  end
until *convergence*;
Build $\boldsymbol{Tr}$ according to $\mathbf{M}$ and $\boldsymbol{\phi}$.

4 Experiments

4.1 Datasets

We conduct experiments on four widely used benchmark datasets: 20NEWS (Miao et al., 2017), Reuters (Wu et al., 2020), Wikitext-103 (Nan et al., 2019), and Rcv1-v2 (Miao et al., 2017). 20NEWS and Reuters are two news corpora. Wikitext-103 is a language modeling dataset extracted from Wikipedia, and Rcv1-v2 is a large version of Reuters. Table 1 presents the statistics of these datasets, where the vocabulary is obtained by following the same preprocessing steps in the original paper. For each corpus, we randomly select 5% of training samples as the validation set.

Dataset	#Docs (Train)	#Docs (Test)	Vocabulary size
20NEWS	11,314	7,531	1,995
Reuters	7,769	3,019	2,000
Wikitext-103	28,472	60	20,000
Rcv1-v2	794,414	10,000	10,000

Table 1: The statistics of datasets.

4.2 Experimental Setup

For tree-structured topic models, we adopt hLDA (Griffiths et al., 2004)[3], rCRP (Kim et al., 2012),

[3]Note that hLDA was named as nCRP (Blei et al., 2010) in (Isonuma et al., 2020).

and TSNTM (Isonuma et al., 2020) as our baselines. For all these models, the max-depth of topic tree is set to 3 by following (Isonuma et al., 2020).

For nonparametric or flat topic models, we adopt HDP (Teh et al., 2005), GSM & GSB (Miao et al., 2017), NB-NTM & GNB-NTM (Wu et al., 2020), and iTM-VAE & HiTM-VAE (Ning et al., 2020) as baselines. HDP is a classical nonparametric topic model that allows potentially an infinite number of topics. GSM & GSB are two NVI-based models using Gaussian priors. In particular, GSB uses Gaussian distributions to generate stick-breaking fractions. NB-NTM & GNB-NTM are two flat neural topic models based on Negative Binomial and Gamma Negative Binomial processes respectively. For iTM-VAE & HiTM-VAE, they extended the method in (Nalisnick and Smyth, 2017) to introduce nonparametric processes into the NVI framework by extracting the potential infinite topics.

We directly use the publicly available codes of hLDA[4], rCRP[5], TSNTM[6], HDP[7], NB-NTM & GNB-NTM[8], and iTM-VAE & HiTM-VAE[9]. Besides, we implement GSM & GSB based on the original paper. For all parametric models, the number of topics is set to 50 and 200 as in (Miao et al., 2017). For nonparametric models based on SBP, the truncation level is set to 200, and the concentration parameter β_0 for the GEM distribution is chosen from [5, 10, 15, 20, 25, 30] using each validation set. In particular, we sequentially choose the topics, of which the sum of probabilities in the whole corpus exceeds 95%, as the active ones. For neural baselines and our proposed model, we set the size of hidden layers to 256 and use one sample for NVI by following (Miao et al., 2017).

All the experiments are conducted on a workstation in Python/Java environment equipped with 40G memory. In the following, we do not report the results of hLDA and rCRP on Rcv1-v2 since they failed to achieve convergence in 48 hours.

4.3 Topic Hierarchy Analysis

As mentioned in (Viegas et al., 2020), a reasonable topic hierarchy means that topics near the root should be more general while the ones close to the leaves should be more specific. To this end, we adopt topic specialization (Kim et al., 2012) as an indicator for the evaluation of topical hierarchy. The specialization of a topic is the cosine distance between the word distribution of the topic and the term frequency vector of the entire corpus. A higher specialization score implies that the topic is more specialized. Figure 4 presents the average topic specialization scores of each level for different tree-structured models. The results indicate that nTSNTM and rCRP can achieve a reasonable pattern of topic specialization at different levels, i.e., the scores become higher as the level becomes deeper. We also observe that the baseline of TSNTM generates more specific topics at the second level than the third level, which indicates an unreasonable topic hierarchy. For the baseline of hLDA, there is a leap of topic specialization from level 2 to level 3, especially for 20NEWS. The reason may be that each document is generated by topics along a single path for hLDA, which renders the large specialization of the topics at level 3 since they are all restricted to one topic from level 2.

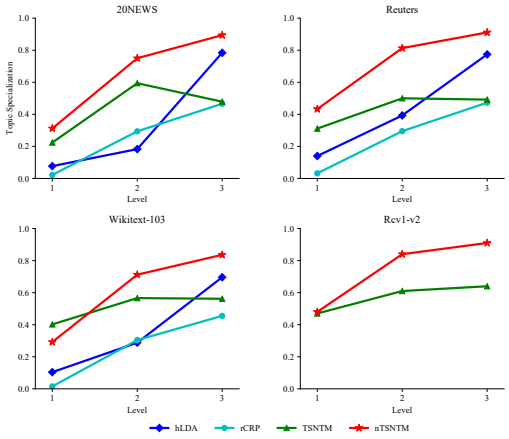

Figure 4: Topic specialization of different tree-structured topic models at each level.

A reasonable topic hierarchy also indicates that child topics are coherent with their corresponding parent topics (Viegas et al., 2020). To measure the relations of two connected topics, we develop a new metric named cross-level NPMI (CLNPMI) to measure the relations of two connected topics by calculating the average NPMI value of every two different topic words from a parent topic and its child. In the above, NPMI was proposed by Lau et al. (2014) which evaluates the relation between

[4] https://github.com/joewandy/hlda
[5] https://github.com/uilab-github/rCRP
[6] https://github.com/misonuma/tsntm
[7] https://github.com/arnim/HDP
[8] https://github.com/mxiny/NB-NTM
[9] https://github.com/walkerning/itmvae_public

two words w_i and w_j as follows:

$$\text{NPMI}(w_i, w_j) = \frac{\log \frac{P(w_i, w_j)}{P(w_i)P(w_j)}}{-\log(P(w_i, w_j))}. \quad (15)$$

Based on NPMI, we define CLNPMI as:

$$\text{CLNPMI}(W_p, W_c) = \frac{1}{|W'_p||W'_c|} \sum_{w_i \in W'_p} \sum_{w_j \in W'_c} \text{NPMI}(w_i, w_j), \quad (16)$$

where $W'_p = W_p - W_c$ and $W'_c = W_c - W_p$, in which, W_p and W_c denote the top N words of a parent topic and one of its children. To avoid degenerating into NPMI when the parent and the child topics are highly similar, CLNPMI is estimated by the distinct words between every two topics.

To evaluate the topic redundancy for a tree, we introduce a new measurement named the averaged overlap rate (OR) and adopt the widely-used topic uniqueness (TU) (Nan et al., 2019). OR measures the averaged repetition ratio of top N words between parent topics and their children, which is defined as: $\text{OR}(W_p, W_c) = \frac{|W_p \cap W_c|}{N}$. TU calculates the uniqueness of all topics by $\text{TU} = \frac{1}{K}\sum_{k=1}^{K} \text{TU}(k)$, where K is the number of topics and $\text{TU}(k)$ is defined as:

$$\text{TU}(k) = \frac{1}{N} \sum_{n=1}^{N} \frac{1}{cnt(n, k)}. \quad (17)$$

In the above, $cnt(n, k)$ is the total number of times the n^{th} top word in topic k appears in the top N words across all topics.

Model		hLDA	rCRP	TSNTM	nTSNTM
20NEWS	CLNPMI (↑)	0.065	0.098	0.086	**0.122**
	TU (↑)	0.051	0.285	0.430	**0.760**
	OR (↓)	0.056	0.404	0.083	**0.053**
Reuters	CLNPMI (↑)	0.050	0.072	0.027	**0.102**
	TU (↑)	0.447	0.227	0.370	**0.708**
	OR (↓)	0.105	0.515	0.176	**0.066**
Wikitext-103	CLNPMI (↑)	0.063	0.088	0.065	**0.113**
	TU (↑)	0.597	0.355	0.615	**0.730**
	OR (↓)	0.087	0.447	0.078	**0.069**
Rcv1-v2	CLNPMI (↑)	–	–	0.028	**0.088**
	TU (↑)	–	–	0.544	**0.802**
	OR (↓)	–	–	0.051	**0.042**

Table 2: CLNPMI, TU, and OR scores of tree-structured topic models, in which, higher CLNPMI and TU with a lower OR indicate better performance. The best value on each metric is highlighted by boldface.

For each of the aforementioned metrics, we calculate the average scores of 5, 10, and 15 top words. Table 2 shows the performance of different models, where each method is run for 5 times and the average values are presented. The results indicate that our model significantly outperforms the baselines in most cases, with p-values less than 0.05. For hLDA and our nTSNTM on the 20NEWS dataset, the difference is not statistically significant on the OR metric, with a p-value equal to 0.391. This validates the effectiveness of the bottom-up structure for nTSNTM, in which, non-leaf topics are activated when their offsprings are chosen.

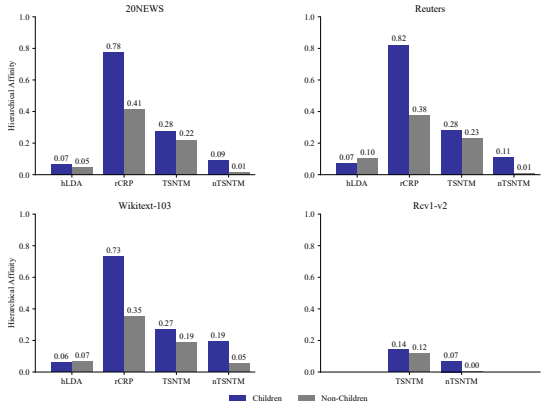

Figure 5: Hierarchical affinity scores.

We also present the hierarchical affinity (Kim et al., 2012) for each model to measure whether the parent topic is more similar to its child topics than the descendants of other parent topics. The average cosine similarities of the parent topic's word distribution to children topics and non-children topics are shown in Figure 5. For parent topics, both rCRP and nTSNTM clearly show stronger affinities with children topics than non-children topics. But rCRP suffers from the high redundancy, which can be indicated by the high similarities ($0.73 \sim 0.82$) between parent topics and sub-topics. To intuitively demonstrate the ability of our model in generating a topic tree, we present several topics extracted from 20NEWS by our nTSNTM and the existing NVI-based TSNTM in Figures 6 and 7, respectively. The results indicate that our model is able to learn a reasonable tree-structured topic hierarchy with low redundancy. While for TSNTM, we notice that there is a low degree of discrimination between topics at the second and the third levels. In addition, topics of the same group at the third level are highly repetitive, including "rec.sport.baseball" and "talk.politics.misc". For completeness, we further check topics extracted from 20NEWS by hLDA and rCRP. The results indicate that each topic at the second level is too general to represent

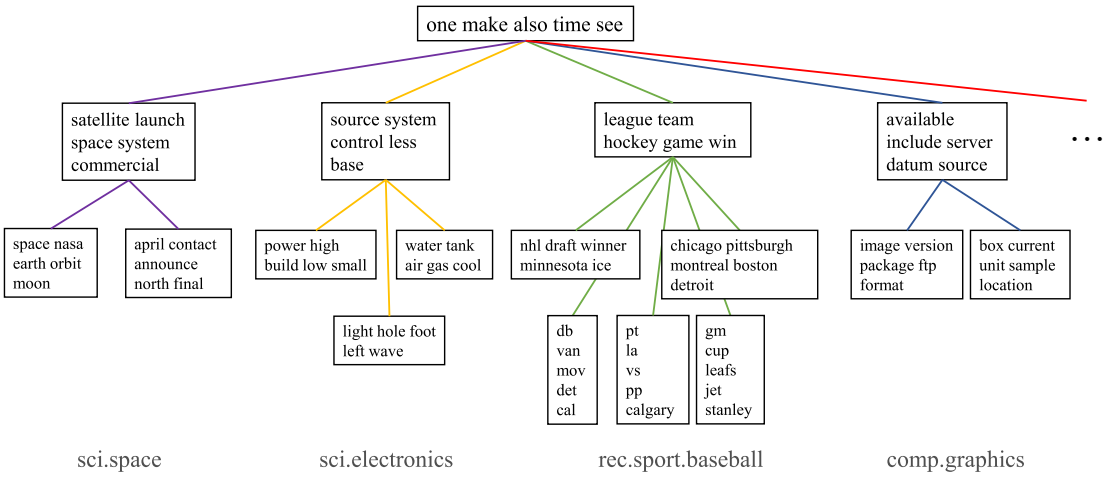

Figure 6: Topic samples extracted from 20NEWS by nTSNTM, where top 5 words are listed for each topic.

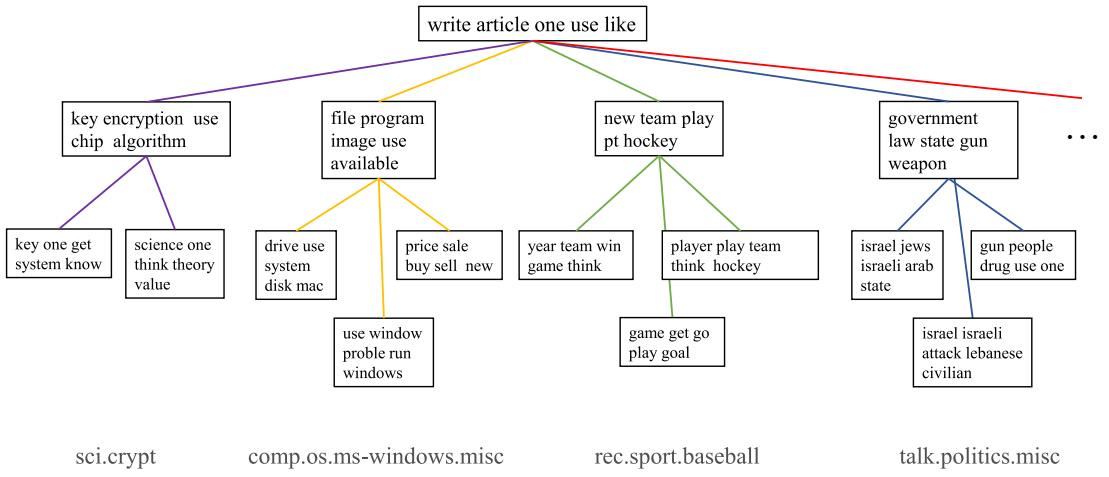

Figure 7: Topic samples extracted from 20NEWS by TSNTM, where top 5 words are listed for each topic.

a topic branch and the affiliations are unclear for hLDA. Although rCRP can generate meaningful topics with appropriate affiliations between different levels, it suffers from a high topic redundancy.

4.4 Comparison on Topic Interpretability

In this part, we use the widely adopted NPMI (Miao et al., 2017; Liu et al., 2019; Wu et al., 2020; Ning et al., 2020; Isonuma et al., 2020) to evaluate topic interpretability[10]. As mentioned in (Lau et al., 2014), the NPMI is a measurement of topic coherence which closely corresponds to the ranking of topic interpretability by human annotators. Table 3 shows the NPMI of 50 and 200 topics for parametric topic models and topics induced automatically for nonparametric topic models. We run each model for 5 times and present the average results. Firstly, nTSNTM outperforms all tree-structured baselines, and the difference is statistically significant at the level of 0.05 (except for TSNTM on the Rcv1-v2 dataset). Secondly, nTSNTM shows competitive performance when compared with the best flat baselines. In particular, except for HiTM-VAE on the Reuters dataset, the results of all the other top-performing baselines are not significantly better than those of our model.

4.5 Evaluating Data Scalability

To evaluate data scalability, we randomly sample several numbers of documents (12.5k, 25k, 50k, 100k, 200k, 400k, and all) from the training

[10] We do not estimate the perplexity for the following two reasons. First, the perplexity of sampling-based and NVI-based models is difficult to compare directly (Isonuma et al., 2020). Second, the prior of NVI-based methods has a large influence on the perplexity since the KL-divergence may vary greatly for different priors (Burkhardt and Kramer, 2019).

Model	20NEWS NPMI (↑)		Reuters NPMI (↑)		Wikitext-103 NPMI (↑)		Rcv1-v2 NPMI (↑)	
	50	200	50	200	50	200	50	200
GSM	0.211	0.165	0.198	0.155	0.214	0.217	**0.231**	0.062
GSB	0.231	0.191	0.152	0.136	0.229	0.131	0.226	0.121
NB-NTM	0.188	0.223	0.248	0.245	0.127	0.125	0.151	0.187
GNB-NTM	**0.240**	0.228	0.237	0.255	0.127	0.093	0.163	0.191
HDP	0.192		0.266		0.157		0.178	
iTM-VAE	0.195		0.201		0.184		0.161	
HiTM-VAE	0.237		**0.269**		0.233		0.179	
rCRP	0.186		0.206		0.201		–	
hLDA	0.221		0.185		0.186		–	
TSNTM	0.212		0.206		0.213		0.225	
nTSNTM	0.237		0.234		**0.237**		0.224	

Table 3: NPMI of each model, where the best result is marked in bold. The topic numbers of parametric models are set to 50 and 200, and those of nonparametric models are automatically determined.

set of Rcv1-v2 to run our model and other tree-structured baselines. The sampling-based models (i.e., hLDA and rCRP) are run on an Intel Xeon Skylake 6133 CPU with 8 cores, and NVI-based models (i.e., TSNTM and nTSNTM) are tested on an Nvidia Tesla V100 GPU. Figure 8 shows the training time of these topic models. Our nTSNTM shows an advantage in data scalability when compared with baselines. Although TSNTM is also scalable to a large corpus by GPU acceleration, it applies a doubly-recurrent network which largely slows down the model speed. hLDA and rCRP spend considerable computation time on path sampling, which is much more serious when dealing with a large-scale dataset. Additionally, these two sampling-based models are serial, which means they can only utilize one core of the CPU.

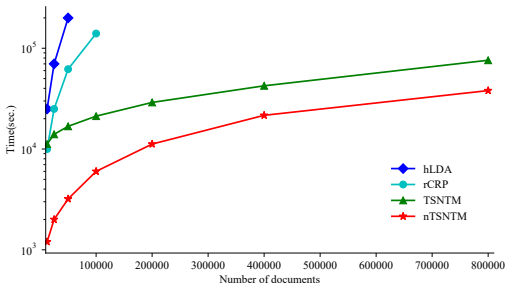

Figure 8: Training time of different models on various numbers of documents. The curves of hLDA and rCRP are incomplete because the time they cost is not comparable. Particularly, it costs over 48 hours for training when the numbers of documents are larger than 50k and 100k for hLDA and rCRP, respectively.

4.6 Impact of the Concentration Parameter

We further validate the nonparametric property of our model. Figure 9 shows the impact of β_0 on the number of active topics. Firstly, we can see that the topic numbers of all models grow when increasing β_0. The reason is that β_0 controls the smoothness of SBP, and that a larger value leads to a smoother degree, i.e., more topics. Secondly, compared with iTM-VAE and HiTM-VAE, the number of topics found by nTSNTM is closer to the one extracted by HDP, which demonstrates that our model is able to approximate the nonparametric property of HDP.

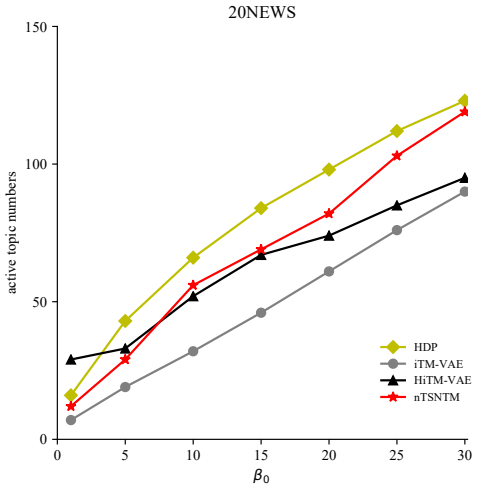

Figure 9: Active topic numbers of different models with various values of β_0.

5 Conclusion

In this paper, we propose a nonparametric tree-structured neural topic model named nTSNTM. Our method explicitly models the dependency of latent variables from different layers, and combines them to reconstruct the input text. By coupling SBP with dependency matrices, we can update the tree structure automatically. Extensive experiments validate the effectiveness of our nTSNTM on generating a reasonable topic tree with low topic redundancies. Furthermore, our model can be trained 2 times faster than the existing NVI-based TSNTM with approximately 800k documents. In the future, we plan to apply our method to aspect extraction.

Acknowledgment

We are grateful to the reviewers for their constructive comments and suggestions on this study. This work has been supported by the National Natural Science Foundation of China (61972426) and Guangdong Basic and Applied Basic Research Foundation (2020A1515010536).

References

David Alvarez-Melis and T. Jaakkola. 2017. Tree-structured decoding with doubly-recurrent neural networks. In *Proceedings of the 5th International Conference on Learning Representations*.

David M. Blei, Thomas L. Griffiths, and Michael I. Jordan. 2010. The nested chinese restaurant process and bayesian nonparametric inference of topic hierarchies. *Journal of the ACM*, 57(2):1–30.

David M. Blei, Andrew Y. Ng, and Michael I. Jordan. 2003. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3(1):993–1022.

Sophie Burkhardt and Stefan Kramer. 2019. Decoupling sparsity and smoothness in the dirichlet variational autoencoder topic model. *Journal of Machine Learning Research*, 20(131):1–27.

Zoubin Ghahramani, Michael Jordan, and Ryan P Adams. 2010. Tree-structured stick breaking for hierarchical data. In *Advances in Neural Information Processing Systems*, pages 19–27.

Thomas L. Griffiths, Michael I. Jordan, Joshua Tenenbaum, and David M. Blei. 2004. Hierarchical topic models and the nested chinese restaurant process. In *Advances in Neural Information Processing Systems*, pages 17–24.

Geoffrey E. Hinton, Oriol Vinyals, and Jeffrey Dean. 2015. Distilling the knowledge in a neural network. *CoRR*, abs/1503.02531.

Hemant Ishwaran and Lancelot F. James. 2001. Gibbs sampling methods for stick-breaking priors. *Journal of the American Statistical Association*, 96(453):161–173.

Masaru Isonuma, Junichiro Mori, Danushka Bollegala, and Ichiro Sakata. 2020. Tree-structured neural topic model. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 800–806.

Joon Hee Kim, Dongwoo Kim, Suin Kim, and Alice H. Oh. 2012. Modeling topic hierarchies with the recursive chinese restaurant process. In *Proceedings of the 21st ACM International Conference on Information and Knowledge Management*, pages 783–792.

Suin Kim, Jianwen Zhang, Zheng Chen, Alice H. Oh, and Shixia Liu. 2013. A hierarchical aspect-sentiment model for online reviews. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 526–533.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *Proceedings of the 3rd International Conference on Learning Representations*.

Diederik P. Kingma and Max Welling. 2014. Auto-encoding variational bayes. In *Proceedings of the 2nd International Conference on Learning Representations*.

P. Kumaraswamy. 1980. A generalized probability density function for double-bounded random processes. *Journal of Hydrology*, 46:79–88.

Jey Han Lau, David Newman, and Timothy Baldwin. 2014. Machine reading tea leaves: Automatically evaluating topic coherence and topic model quality. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 530–539.

Wei Li and Andrew McCallum. 2006. Pachinko allocation: Dag-structured mixture models of topic correlations. In *Proceedings of the 23rd International Conference on Machine Learning*, pages 577–584.

Luyang Liu, Heyan Huang, Yang Gao, Yongfeng Zhang, and Xiaochi Wei. 2019. Neural variational correlated topic modeling. In *Proceeding of The World Wide Web Conference*, pages 1142–1152.

Rui Liu, Xingguang Wang, Deqing Wang, Yuan Zuo, He Zhang, and Xianzhu Zheng. 2018. Topic splitting: A hierarchical topic model based on non-negative matrix factorization. *Journal of Systems Science and Systems Engineering*, 27:479–496.

Tengfei Liu, Nevin L Zhang, and Peixian Chen. 2014. Hierarchical latent tree analysis for topic detection. In *Proceedings of the 2014 European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 256–272.

Yishu Miao, Edward Grefenstette, and Phil Blunsom. 2017. Discovering discrete latent topics with neural variational inference. In *Proceedings of the 34th International Conference on Machine Learning*, pages 2410–2419.

David Mimno, Wei Li, and Andrew McCallum. 2007. Mixtures of hierarchical topics with pachinko allocation. In *Proceedings of the 24th International Conference on Machine Learning*, pages 633–640.

Zhao-Yan Ming, Kai Wang, and Tat-Seng Chua. 2010. Prototype hierarchy based clustering for the categorization and navigation of web collections. In *Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval*, pages 2–9.

Eric T. Nalisnick and Padhraic Smyth. 2017. Stick-breaking variational autoencoders. In *Proceedings of the 5th International Conference on Learning Representations*.

Feng Nan, Ran Ding, Ramesh Nallapati, and Bing Xiang. 2019. Topic modeling with wasserstein autoencoders. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6345–6381.

Xuefei Ning, Yin Zheng, Zhuxi Jiang, Yu Wang, Huazhong Yang, Junzhou Huang, and Peilin Zhao. 2020. Nonparametric topic modeling with neural inference. *Neurocomputing*, 399:296–306.

John Paisley, Chong Wang, David M. Blei, and Michael I. Jordan. 2014. Nested hierarchical dirichlet processes. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(2):256–270.

Jim Pitman. 2006. Combinatorial stochastic processes. In *Technical Report 621, Dept. Statistics, UC Berkeley*.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. 2014. Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the 31st International Conference on Machine Learning*, pages 1278–1286.

Jayaram Sethuraman. 1994. A constructive definition of dirichlet priors. *Statistica Sinica*, pages 639–650.

Yee Teh, Michael Jordan, Matthew Beal, and David Blei. 2005. Sharing clusters among related groups: Hierarchical dirichlet processes. In *Advances in Neural Information Processing Systems*, pages 1385–1392.

Felipe Viegas, Washington Cunha, Christian Gomes, Antônio Pereira, Leonardo C. da Rocha, and Marcos André Gonçalves. 2020. Cluhtm - semantic hierarchical topic modeling based on cluwords. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8138–8150.

Chong Wang and David M. Blei. 2009. Variational inference for the nested chinese restaurant process. In *Advances in Neural Information Processing Systems*, pages 1990–1998.

Jiemin Wu, Yanghui Rao, Zusheng Zhang, Haoran Xie, Qing Li, Fu Lee Wang, and Ziye Chen. 2020. Neural mixed counting models for dispersed topic discovery. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6159–6169.

Yueshen Xu, Jianwei Yin, Jianbin Huang, and Yuyu Yin. 2018. Hierarchical topic modeling with automatic knowledge mining. *Expert Systems with Applications*, 103:106–117.