Recurrent Hierarchical Topic-Guided RNN for Language Generation

Dandan Guo [1] **Bo Chen** [1] **Ruiying Lu** [1] **Mingyuan Zhou** [2]

Abstract

To simultaneously capture syntax and global semantics from a text corpus, we propose a new larger-context recurrent neural network (RNN) based language model, which extracts recurrent hierarchical semantic structure via a dynamic deep topic model to guide natural language generation. Moving beyond a conventional RNN-based language model that ignores long-range word dependencies and sentence order, the proposed model captures not only intra-sentence word dependencies, but also temporal transitions between sentences and inter-sentence topic dependencies. For inference, we develop a hybrid of stochastic-gradient Markov chain Monte Carlo and recurrent autoencoding variational Bayes. Experimental results on a variety of real-world text corpora demonstrate that the proposed model not only outperforms larger-context RNN-based language models, but also learns interpretable recurrent multilayer topics and generates diverse sentences and paragraphs that are syntactically correct and semantically coherent.

1. Introduction

Both topic and language models are widely used for text analysis. Topic models, such as latent Dirichlet allocation (LDA) (Blei et al., 2003; Griffiths & Steyvers, 2004; Hoffman et al., 2013) and its nonparametric Bayesian generalizations (Teh et al., 2006; Zhou & Carin, 2015), are well suited for extracting document-level word concurrence patterns into latent topics from a text corpus. Their modeling power has been further enhanced by introducing multilayer deep representation (Srivastava et al., 2013; Mnih & Gregor, 2014; Gan et al., 2015; Zhou et al., 2016; Zhao et al., 2018; Zhang et al., 2018). While having semantically meaningful latent representation, they typically treat each document as a bag of words (BoW), ignoring word order (Griffiths et al., 2004; Wallach, 2006). To take the word order into consideration, Wang et al. (2019a) introduce a customized convolutional operator and probabilistic pooling into a topic model, which successfully captures local dependencies and forms phrase-level topics but has limited ability in modeling sequential dependencies and generating word sequences.

Language models have become key components of various natural language processing tasks, such as text summarization (Rush et al., 2015; Gehrmann et al., 2018), speech recognition (Mikolov et al., 2010; Graves et al., 2013), machine translation (Sutskever et al., 2014; Cho et al., 2014), and image captioning (Vinyals et al., 2015; Mao et al., 2015; Xu et al., 2015; Gan et al., 2017; Rennie et al., 2017; Fan et al., 2020). The primary purpose of a language model is to capture the distribution of a word sequence, commonly with a recurrent neural network (RNN) (Mikolov et al., 2011; Graves, 2013) or a Transformer-based model (Vaswani et al., 2017; Dai et al., 2019; Devlin et al., 2019; Radford et al., 2018; 2019). In this paper, utilizing a deep dynamic model for sequentially observed count vectors and introducing a recurrent variational inference network, we focus on improving RNN-based language models that often have much fewer parameters and are easier to perform end-to-end training.

While RNN-based language models do not ignore word order, they often assume that the sentences of a document are independent of each other. This simplifies the modeling task to independently assigning probabilities to individual sentences, ignoring their order and document context (Tian & Cho, 2016). Such language models may consequently fail to capture the long-range dependencies and global semantic meaning of a document (Dieng et al., 2017; Wang et al., 2018). While a naive solution to explore richer contextual information is to concatenate all previous sentences into a single "sentence" and use it as the input of an RNN-based language model, in practice, the length of that sentence is limited given the constraint of memory and computation resource. Even if making the length very long, this naive solution rarely works well enough to satisfactorily address the long-standing research problem of capturing long-range dependencies, motivating a variety of more sophisticated methods to improve existing language models (Dieng et al., 2017; Lau et al., 2017; Wang et al., 2018; 2019b; Dai et al.,

[1] National Laboratory of Radar Signal Processing, Xidian University, Xi'an, China. [2] McCombs School of Business, The University of Texas at Austin, Austin, TX 78712, USA. Correspondence to: Bo Chen <bchen@mail.xidian.edu.cn>.

Proceedings of the 37^{th} International Conference on Machine Learning, Online, PMLR 119, 2020. Copyright 2020 by the author(s).

2019). Moreover, such a solution often clearly enlarges the model size, increasing the difficulty of optimization and risk of overfitting (Dieng et al., 2017). Finding better ways to model long-range dependencies in language modeling is therefore an open research challenge. To relax the sentence independence assumption in language modeling, Tian & Cho (2016) propose larger-context language models that model the context of a sentence by representing its preceding sentences as either a single or a sequence of BoW vectors, which are then fed directly into the sentence modeling RNN.

Since topic models are well suited for capturing long-range dependencies, an alternative approach attracting significant recent interest is leveraging topic models to improve RNN-based language models. Mikolov & Zweig (2012) use pre-trained topic model features as an additional input to the RNN hidden states and/or output. Dieng et al. (2017) and Ahn et al. (2017) combine the predicted word distributions, given by both a topic model and a language model, under variational autoencoder (Kingma & Welling, 2013). Lau et al. (2017) introduce an attention based convolutional neural network to extract semantic topics, which are used to extend the RNN cell. Wang et al. (2018) learn the global semantic coherence of a document via a neural topic model and use the learned latent topics to build a mixture-of-experts language model. Wang et al. (2019b) further specify a Gaussian mixture model as the prior of the latent code in variational autoencoder, where each mixture component corresponds to a topic.

While clearly improving the performance of the end task, these existing topic-guided methods still have clear limitations. For example, they only utilize shallow topic models with only a single stochastic hidden layer for data generation. Note several neural topic models use deep neural networks to construct their variational encoders, but still use shallow generative models as decoders (Miao et al., 2017; Srivastava & Sutton, 2017). Another key limitation lies in ignoring the sentence order, as each document is treated as a bag of sentences. Thus once the topic weight vector learned from the document context is given, the task is often reduced to independently assigning probabilities to individual sentences (Lau et al., 2017; Wang et al., 2018; 2019b).

In this paper, as depicted in Fig. 1, we propose to use recurrent gamma belief network (rGBN) to guide a stacked RNN for language modeling. We refer to the model as rGBN-RNN, which integrates rGBN (Guo et al., 2018), a deep recurrent topic model, and stacked RNN (Graves, 2013; Chung et al., 2017), a neural language model, into a novel larger-context RNN-based language model. It simultaneously learns a deep recurrent topic model, extracting document-level multi-layer word concurrence patterns and sequential topic weight vectors for sentences, and an expressive language model, capturing both short- and long-range word sequential dependencies. For inference,we equip rGBN-RNN (decoder) with a novel recurrent variational inference network (encoder), and train it end-to-end by maximizing an evidence lower bound (ELBO). Different from the stacked RNN based language model in Chung et al. (2017), which relies on three types of customized training operations (UPDATE, COPY, FLUSH) to extract multi-scale structures, the language model in rGBN-RNN learns such structures purely in a data driven manner, under the guidance of the temporally and hierarchically connected stochastic layers of rGBN. Note while both rGBN and stacked-RNN are existing methods, integrating them as a larger-context language model involves non-trivial efforts, as we need to not only carefully design how to connect the recurrent hierarchical stochastic layers of rGBN with the deterministic ones of stacked-RNN, but also design a suitable recurrent variational inference network.

The effectiveness of rGBN-RNN as a new larger-context language model is demonstrated both quantitatively, with perplexity and BLEU scores, and qualitatively, with interpretable latent structures and randomly generated sentences and paragraphs. Notably, moving beyond a usual RNN-based language model that generates individual sentences, the proposed rGBN-RNN can generate a paragraph consisting of a sequence of semantically coherent sentences.

2. Recurrent Hierarchical Topic-Guided Language Model

Denote a document of J sentences as $\mathcal{D} = (S_1, S_2, \ldots, S_J)$, where $S_j = (y_{j,1}, \ldots, y_{j,T_j})$ consists of T_j words from a vocabulary of size V. Conventional statistical language models often only focus on the word sequence within a sentence. Assuming that the sentences of a document are independent of each other, they often define

$$\begin{aligned} P(\mathcal{D}) &\approx \textstyle\prod_{j=1}^{J} P\left(S_j\right) \\ &= \textstyle\prod_{j=1}^{J} \left[p\left(y_{j,1}\right) \prod_{t=2}^{T_j} p\left(y_{j,t} \,|\, y_{j,<t}\right)\right]. \end{aligned}$$

RNN-based neural language models define the conditional probability of each word $y_{j,t}$ given all the previous words $y_{j,<t}$ within the sentence S_j, through the softmax function of a hidden state $\boldsymbol{h}_{j,t}$, as

$$\begin{aligned} p\left(y_{j,t} \,|\, y_{j,<t}\right) &= p\left(y_{j,t} \,|\, \boldsymbol{h}_{j,t}\right), \\ \boldsymbol{h}_{j,t} &= f\left(\boldsymbol{h}_{j,<t}, y_{j,t-1}\right), \end{aligned} \tag{1}$$

where $f(\cdot)$ is a non-linear function typically defined as an RNN cell, such as long short-term memory (LSTM) (Hochreiter & Schmidhuber, 1997) and gated recurrent unit (GRU) (Cho et al., 2014).

These RNN-based language models are typically applied only at the word level, without exploiting the document context, and hence often fail to capture long-range dependencies. While Dieng et al. (2017), Lau et al. (2017), and Wang et al. (2018; 2019b) remedy this issue by guiding the language model with a topic model, they still treat a document as a bag of sentences, ignoring sentence order, and lack the ability to extract hierarchical and recurrent topic structures.

We introduce rGBN-RNN, as depicted in Fig. 1 (a), as a new larger-context language model. It consists of two key components: (i) a hierarchical recurrent topic model (rGBN), and (ii) a stacked RNN-based language model. We use rGBN to capture both global semantics across documents and long-range inter-sentence dependencies within a document, and use the language model to learn the local syntactic relationships between the words within a sentence. Similar to Lau et al. (2017) and Wang et al. (2018), we represent a document as a sequence of sentence-context pairs as $(\{S_1, \boldsymbol{d}_1\}, \dots, \{S_J, \boldsymbol{d}_J\})$, where $\boldsymbol{d}_j \in \mathbb{Z}_+^{V_c}$ summarizes the document excluding S_j, specifically $(S_1, ..., S_{j-1}, S_{j+1}, ..., S_J)$, into a BoW count vector, with V_c denoting the size of the vocabulary excluding stop words. During testing, we redefine $\boldsymbol{d}_j$ as the BoW vector summarizing only the preceding sentences, $i.e.$, $S_{1:j-1}$, which will be further clarified when presenting experimental results. Note a naive way to utilize sentence order is to treat each sentence as a document, use a dynamic topic model (Blei & Lafferty, 2006) to capture the temporal dependencies of the latent topic-weight vectors, each of which is fed to the RNN to model the word sequence of its corresponding sentence. However, the sentences are often too short to be well modeled by a topic model. In our setting, as $\boldsymbol{d}_j$ summarizes the document-level context of S_j, it is in general sufficiently long for topic modeling.

2.1. Hierarchical Recurrent Topic Model

As shown in Fig. 1 (a), to model the time-varying sentence-context count vectors $\boldsymbol{d}_j$ in document $\mathcal{D}$, the generative process of the rGBN component, from the top to bottom layers, is expressed as

$$\begin{aligned} \boldsymbol{\theta}_j^L &\sim \text{Gam}\left(\boldsymbol{\Pi}^L \boldsymbol{\theta}_{j-1}^L,\ \tau_0\right), \cdots, \\ \boldsymbol{\theta}_j^l &\sim \text{Gam}\left(\boldsymbol{\Phi}^{l+1} \boldsymbol{\theta}_j^{l+1} + \boldsymbol{\Pi}^l \boldsymbol{\theta}_{j-1}^l,\ \tau_0\right), \cdots, \\ \boldsymbol{\theta}_j^1 &\sim \text{Gam}\left(\boldsymbol{\Phi}^2 \boldsymbol{\theta}_j^2 + \boldsymbol{\Pi}^1 \boldsymbol{\theta}_{j-1}^1,\ \tau_0\right), \\ \boldsymbol{d}_j &\sim \text{Pois}\left(\boldsymbol{\Phi}^1 \boldsymbol{\theta}_j^1\right), \end{aligned} \tag{2}$$

where $\boldsymbol{\theta}_j^l \in \mathbb{R}_+^{K_l}$ denotes the gamma distributed topic weight vector of sentence j at layer $l \in \{1, \dots, L\}$, $\boldsymbol{\Pi}^l \in \mathbb{R}_+^{K_l \times K_l}$ the transition matrix of layer l that captures cross-topic temporal dependencies, $\boldsymbol{\Phi}^l \in \mathbb{R}_+^{K_{l-1} \times K_l}$ the loading matrix at layer l, K_l the number of topics of layer l, and $\tau_0 \in \mathbb{R}_+$ a scaling hyperparameter. At $j = 1$, $\boldsymbol{\theta}_1^l \sim \text{Gam}\left(\boldsymbol{\Phi}^{l+1} \boldsymbol{\theta}_1^{l+1}, \tau_0\right)$ for $l = 1, \dots, L-1$ and $\boldsymbol{\theta}_1^L \sim \text{Gam}\left(\nu, \tau_0\right)$, where $\nu = \mathbf{1}_{K_L}$. Following Guo et al. (2018) and Zhou et al. (2015), the Dirichlet priors are placed on the columns of $\boldsymbol{\Pi}^l$ and $\boldsymbol{\Phi}^l$, $i.e.$, $\boldsymbol{\pi}_k^l$ and $\boldsymbol{\phi}_k^l$, which not only makes the latent representation more identifiable and interpretable, but also facilitates inference. The count vector $\boldsymbol{d}_j$ can be factorized into the product of $\boldsymbol{\Phi}^1$ and $\boldsymbol{\theta}_j^1$ under the Poisson likelihood. The shape parameters of $\boldsymbol{\theta}_j^l \in \mathbb{R}_+^{K_l}$ can be factorized into the sum of $\boldsymbol{\Phi}^{l+1} \boldsymbol{\theta}_j^{l+1}$, capturing inter-layer hierarchical dependence, and $\boldsymbol{\Pi}^l \boldsymbol{\theta}_{j-1}^l$, capturing intra-layer temporal dependence. rGBN not only captures the document-level word occurrence patterns inside the training text corpus, but also the sequential dependencies of the sentences inside a document. Note ignoring the recurrent structure, rGBN will reduce to the gamma belief network (GBN) of Zhou et al. (2016), which can be reformulated as a multi-stochastic-layer deep generalization of LDA (Cong et al., 2017a); if setting the number of stochastic hidden layer as $L = 1$, GBN reduces to Poisson factor analysis (Zhou et al., 2012; Zhou & Carin, 2015) . If ignoring its hierarchical structure ($i.e.$, $L = 1$), rGBN reduces to Poisson–gamma dynamical systems of Schein et al. (2016) that generalizes the gamma Markov chain of Acharya et al. (2015) by adding latent state transitions. We refer to the rGBN-RNN without its recurrent structure as GBN-RNN, which no longer models sequential sentence dependencies; see Appendix A for more details.

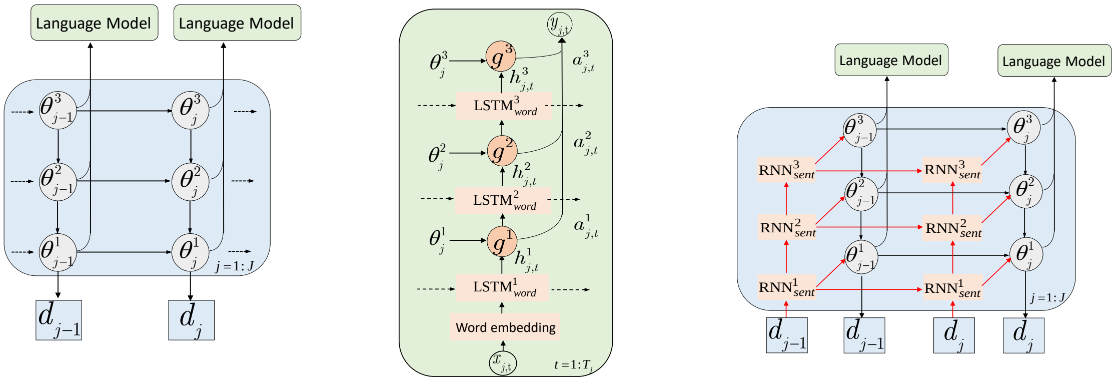
2.2. Language Model

Different from a conventional RNN-based language model, which predicts the next word only using the preceding words within the sentence, we integrate the hierarchical recurrent topic weight vectors $\boldsymbol{\theta}_j^l$ into the language model to predict the word sequence in the jth sentence. Our proposed language model is built upon the stacked RNN proposed in Graves (2013) and Chung et al. (2017), but with the help of rGBN, it no longer requires specialized training heuristics to extract multi-scale latent structures. As shown in Fig. 1 (b), to generate $y_{j,t}$, the t^{th} token of sentence j in a document, we construct the hidden states $\boldsymbol{h}_{j,t}^l$ of the language model, from the bottom to top layers, as

$$\boldsymbol{h}_{j,t}^l = \begin{cases} \text{LSTM}_{\text{word}}^l\left(\boldsymbol{h}_{j,t-1}^l, \boldsymbol{W_e}\left[x_{j,t}\right]\right), & \text{if } l = 1, \\ \text{LSTM}_{\text{word}}^l\left(\boldsymbol{h}_{j,t-1}^l, \boldsymbol{a}_{j,t}^{l-1}\right), & \text{if } 1 < l \leq L, \end{cases} \tag{3}$$

where $\text{LSTM}_{\text{word}}^l$ denotes the word-level LSTM at layer l, $\boldsymbol{W_e}$ are word embeddings to be learned, and $x_{j,t} = y_{j,t-1}$. Note $\boldsymbol{a}_{j,t}^l$ denotes the coupling vector, which combines the temporal topic weight vectors $\boldsymbol{\theta}_j^l$ and hidden output of the word-level LSTM $\boldsymbol{h}_{j,t}^l$ at each time step t. Following Lau et al. (2017), we realize $\boldsymbol{a}_{j,t}^l = g^l\left(\boldsymbol{h}_{j,t}^l, \boldsymbol{\theta}_j^l\right)$ with a gating unit similar to a GRU (Cho et al., 2014), described as

$$\boldsymbol{a}_{j,t}^l = \left(1 - \boldsymbol{z}_{j,t}^l\right) \odot \boldsymbol{h}_{j,t}^l + \boldsymbol{z}_{j,t}^l \odot \hat{\boldsymbol{h}}_{j,t}^l, \tag{4}$$



(a) (b) (c)

Figure 1. (a) The generative model of a three-hidden-layer rGBN-RNN, where the bottom part is the deep recurrent topic model (rGBN), document contexts of consecutive sentences are used as observed data, and upper is the language model. (b) Overview of the language model component, where input $x_{j,t}$ denotes the tth word in jth sentence of a document, $x_{j,t} = y_{j,t-1}$, $\boldsymbol{h}^l_{j,t}$ is the hidden state of the stacked RNN at time step t, and $\boldsymbol{\theta}^l_j$ is the topic weight vector of sentence j at layer l. (c) The overall architecture of the proposed model, including the decoder (rGBN and language model) and encoder (recurrent variational inference network), where the red arrows denote the inference of latent topic weight vectors and the black arrows denote the data generation.

where

$$\boldsymbol{z}^l_{j,t} = \sigma\left(\mathbf{W}^l_z \boldsymbol{\theta}^l_j + \mathbf{U}^l_z \boldsymbol{h}^l_{j,t} + \mathbf{b}^l_z\right),$$
$$\boldsymbol{r}^l_{j,t} = \sigma\left(\mathbf{W}^l_r \boldsymbol{\theta}^l_j + \mathbf{U}^l_r \boldsymbol{h}^l_{j,t} + \mathbf{b}^l_r\right),$$
$$\hat{\boldsymbol{h}}^l_{j,t} = \tanh\left(\mathbf{W}^l_h \boldsymbol{\theta}^l_j + \mathbf{U}^l_h\left(\boldsymbol{r}^l_{j,t} \odot \boldsymbol{h}^l_{j,t}\right) + \mathbf{b}^l_h\right).$$

Denote $\boldsymbol{a}^{1:L}_{j,t}$ as the concatenation of $\boldsymbol{a}^l_{j,t}$ across all layers and $\mathbf{W}_o$ as a weight matrix with V rows; different from (1), the conditional probability of $y_{j,t}$ becomes

$$p\left(y_{j,t} \,|\, y_{j,<t}, \boldsymbol{\theta}^{1:L}_j\right) = \text{softmax}\left(\mathbf{W}_o \boldsymbol{a}^{1:L}_{j,t}\right). \tag{5}$$

There are two main reasons for combining all the latent representations $\boldsymbol{a}^{1:L}_{j,t}$ for language modeling. First, the latent representations exhibit different statistical properties at different stochastic layers of rGBN-RNN, and hence are combined together to enhance their representation power. Second, having "skip connections" from all hidden layers to the output one makes it easier to train the proposed network, reducing the number of processing steps between the bottom of the network and the top and hence mitigating the "vanishing gradient" problem (Graves, 2013).

To sum up, as depicted in Fig. 1 (a), the topic weight vector $\boldsymbol{\theta}^l_j$ of sentence j quantifies the topic usage of its document context $\boldsymbol{d}_j$ at layer l. It is further used as an additional feature of the language model to guide the word generation inside sentence j, as shown in Fig. 1 (b). It is clear that rGBN-RNN has two temporal structures: a deep recurrent topic model to extract the temporal topic weight vectors from the sequential document contexts, and a language model to estimate the probability of each sentence given its corresponding hierarchical topic weight vector. Characterizing the word-sentence-document hierarchy to incorporate both intra- and inter-sentence information, rGBN-RNN learns more coherent and interpretable topics and increases the generative power of the language model. Distinct from existing topic-guided language models, the temporally related hierarchical topics of rGBN exhibit different statistical properties across layers, which helps better guide language model to improve its language generation ability.

2.3. Model Likelihood and Inference

For rGBN-RNN, given $\{\boldsymbol{\Phi}^l, \boldsymbol{\Pi}^l\}^L_{l=1}$, the marginal likelihood of the sequence of sentence-context pairs $(\{\boldsymbol{s}_1, \boldsymbol{d}_1\}, \dots, \{\boldsymbol{s}_J, \boldsymbol{d}_J\})$ of document $\mathcal{D}$ is defined as

$$P\left(\mathcal{D} \,|\, \{\boldsymbol{\Phi}^l, \boldsymbol{\Pi}^l\}^L_{l=1}\right) = \int \prod^J_{j=1} p\left(\boldsymbol{d}_j \,|\, \boldsymbol{\Phi}^1 \boldsymbol{\theta}^1_j\right)$$
$$\left[\prod^{T_j}_{t=1} p\left(y_{j,t} \,|\, y_{j,<t}, \boldsymbol{\theta}^{1:L}_j\right)\right]\left[\prod^L_{l=1} p\left(\boldsymbol{\theta}^l_j \,|\, \boldsymbol{e}^l_j, \tau_0\right)\right] d\boldsymbol{\theta}^{1:L}_{1:J}, \tag{6}$$

where $\boldsymbol{e}^l_j := \boldsymbol{\Phi}^{l+1}\boldsymbol{\theta}^{l+1}_j + \boldsymbol{\Pi}^l \boldsymbol{\theta}^l_{j-1}$. The inference task is to learn the parameters of both the topic model and language model components. One naive solution is to alternate the training between these two components in each iteration: First, the topic model is trained using a sampling based iterative algorithm provided in Guo et al. (2018); Second, the language model is trained with maximum likelihood estimation under a standard cross-entropy loss. While this naive solution can utilize readily available inference algorithms for both rGBN and the language model, it may suffer from stability and convergence issues. Moreover, the need to perform a sampling based iterative algorithm for rGBN inside each iteration limits the scalability of the model for both training and testing.

To this end, we introduce a recurrent variational inference network (encoder) to learn the latent temporal topic weight vectors $\boldsymbol{\theta}^{1:L}_{1:J}$. Denoting $Q = \prod^J_{j=1} \prod^L_{l=1} q(\boldsymbol{\theta}^l_j \,|\, \boldsymbol{d}_j)$, an

ELBO of the log marginal likelihood shown in (6) can be constructed as

$$L = \sum_{j=1}^{J} \sum_{l=1}^{L} \mathbb{E}_Q \left[\ln p \left(\boldsymbol{d}_j \mid \boldsymbol{\Phi}^1 \boldsymbol{\theta}_j^1 \right) + \sum_{t=1}^{T_j} \ln p \left(y_{j,t} \mid y_{j,<t}, \boldsymbol{\theta}_j^{1:L} \right) \right]$$
$$- \sum_{j=1}^{J} \sum_{l=1}^{L} \mathbb{E}_Q \left[\ln \frac{q\left(\boldsymbol{\theta}_j^l \mid \boldsymbol{d}_{\leq j}\right)}{p\left(\boldsymbol{\theta}_j^l \mid \boldsymbol{e}_j^l, \tau_0\right)} \right], \tag{7}$$

which unites both the terms primarily responsible for training the recurrent hierarchical topic model component, and terms for training the RNN language model component. Similar to Zhang et al. (2018), we define $q(\boldsymbol{\theta}_j^l \mid \boldsymbol{d}_j) = \text{Weibull}(\boldsymbol{k}_j^l, \boldsymbol{\lambda}_j^l)$, a random sample from which can be obtained by transforming standard uniform noises $\boldsymbol{\epsilon}_j^l$ as

$$\boldsymbol{\theta}_j^l = \boldsymbol{\lambda}_j^l \left(- \ln(1 - \boldsymbol{\epsilon}_j^l) \right)^{1/\boldsymbol{k}_j^l}. \tag{8}$$

To capture the temporal dependencies between the topic weight vectors, both $\boldsymbol{k}_j^l$ and $\boldsymbol{\lambda}_j^l$, from the bottom to top layers, can be expressed as

$$\boldsymbol{h}_j^{s,l} = \text{RNN}_{\text{sent}}^l \left(\boldsymbol{h}_{j-1}^{s,l}, \boldsymbol{h}_j^{s,l-1} \right),$$
$$\boldsymbol{k}_j^l = f_{\boldsymbol{k}}^l \left(\boldsymbol{h}_j^{s,l} \right), \ \ \boldsymbol{\lambda}_j^l = f_{\boldsymbol{\lambda}}^l \left(\boldsymbol{h}_j^{s,l} \right), \tag{9}$$

where $\boldsymbol{h}_j^{s,0} = \boldsymbol{d}_j$, $\boldsymbol{h}_0^{s,l} = 0$, $\text{RNN}_{\text{sent}}^l$ denotes the sentence-level recurrent encoder at layer l implemented with a basic RNN cell, capturing the sequential relationship between sentences within a document, $\boldsymbol{h}_j^{s,l}$ denotes the hidden state of $\text{RNN}_{\text{sent}}^l$, and superscript s in $\boldsymbol{h}_j^{s,l}$ denotes "sentence-level RNN" used to distinguish the hidden state of language model in (3) . Note both $f_{\boldsymbol{k}}^l$ and $f_{\boldsymbol{\lambda}}^l$ are nonlinear functions mapping state $\boldsymbol{h}_j^{s,l}$ to the parameters of $\boldsymbol{\theta}_j^l$, implemented with $f(\boldsymbol{x}) = \ln(1 + \exp(\mathbf{W}\boldsymbol{x} + \boldsymbol{b}))$.

Rather than finding a point estimate of the global parameters $\{\boldsymbol{\Phi}^l, \boldsymbol{\Pi}^l\}_{l=1}^L$ of the rGBN, we adopt a hybrid inference algorithm by combining TLASGR-MCMC described in Cong et al. (2017a) and Zhang et al. (2018) and our proposed recurrent variational inference network. In other words, the global parameters $\{\boldsymbol{\Phi}^l, \boldsymbol{\Pi}^l\}_{l=1}^L$ can be sampled with TLASGR-MCMC, while the parameters of the language model and recurrent variational inference network, denoted by $\boldsymbol{\Omega}$, can be updated via stochastic gradient descent (SGD) by maximizing the ELBO in (7). We describe a hybrid variational/sampling inference for rGBN-RNN in Algorithm 1 and provide more details about sampling $\{\boldsymbol{\Phi}^l, \boldsymbol{\Pi}^l\}_{l=1}^L$ with TLASGR-MCMC in Appendix B. We defer the details on model complexity to Appendix D.

To sum up, as shown in Fig. 1 (c), the proposed rGBN-RNN works with a recurrent variational autoencoder inference framework, which takes the document context of the jth sentence within a document as input and learns hierarchical topic weight vectors $\boldsymbol{\theta}_j^{1:L}$ that evolve sequentially with j. The learned topic vectors in different layer are then used to reconstruct the document context input and as an additional feature for the language model to generate the jth sentence.

Algorithm 1 Hybrid TLASGR-MCMC and recurrent autoencoding variational inference for rGBN-RNN.

```
Set mini-batch size $m$ and the number of layer $L$
Initialize encoder and neural language model parameters $\boldsymbol{\Omega}$, and topic model parameters $\{\boldsymbol{\Phi}^l, \boldsymbol{\Pi}^l\}_{l=1}^L$.
for $iter = 1, 2, \cdots$ do
    Randomly select a mini-batch of $m$ documents consisting of $J$ sentences to form a subset $\mathbf{X} = \{\boldsymbol{d}_{i,1:J}, \boldsymbol{s}_{i,1:J}\}_{i=1}^m$;
    Draw random noise $\left\{\boldsymbol{\epsilon}_{i,j}^l\right\}_{i=1,j=1,l=1}^{m,J,L}$ from uniform distribution;
    Calculate $\nabla_{\boldsymbol{\Omega}} L \left(\boldsymbol{\Omega}, \boldsymbol{\Phi}^l, \boldsymbol{\Pi}^l; \mathbf{X}, \boldsymbol{\epsilon}_{i,j}^l\right)$ according to (7), and update $\boldsymbol{\Omega}$; Sample $\boldsymbol{\theta}_{i,j}^l$ from (8) and (9) via $\boldsymbol{\Omega}$ to update $\{\boldsymbol{\Pi}^l\}_{l=1}^L$ and $\{\boldsymbol{\Phi}^l\}_{l=1}^L$, as described in Appendix B;
end for
```

3. Experimental Results

We consider three publicly available corpora, including APNEWS, IMDB, and BNC. The links, preprocessing steps, and summary statistics for them are deferred to Appendix C. We consider a recurrent variational inference network for rGBN-RNN to infer $\boldsymbol{\theta}_j^l$, as shown in Fig. 1 (c), whose number of hidden units in (9) are set the same as the number of topics at the corresponding layer. Following Lau et al. (2017), word embeddings are pre-trained 300-dimension word2vec Google News vectors (https://code.google.com/archive/p/word2vec/). Dropout with a rate of 0.4 is used to the input of the stacked-RNN at each layer, *i.e.*, $\boldsymbol{a}_{j,t}^l$ or $\boldsymbol{W}_{\boldsymbol{e}}\left[x_{j,t}\right]$ in (3). The gradients are clipped if the norm of the parameter vector exceeds 5. We use the Adam optimizer (Kingma & Ba, 2015) with learning rate 10^{-3}. The length of an input sentence is fixed to 30. We set the mini-batch size as 8, number of training epochs as 5, and τ_0 as 1. Python (TensorFlow) code is provided at `https://github.com/Dan123dan/rGBN-RNN`

3.1. Quantitative Comparison

Perplexity: For fair comparison, we use standard language model perplexity as the evaluation metric. We consider the following baselines: *1)* A standard LSTM language model (Hochreiter & Schmidhuber, 1997); *2)* LCLM (Tian & Cho, 2016), a larger-context language model that incorporates context from preceding sentences, which are treated as a bag of words; *3)* A standard LSTM language model incorporating the topic information of a separately trained LDA (LDA+LSTM); *4)* Topic-RNN (Dieng et al., 2017), a hybrid model rescoring the prediction of the next word by incorporating the topic information through a linear transformation; *5)* TDLM (Lau et al., 2017), a joint learning framework that learns a convolution based topic model and a language model simultaneously; *6)* TCNLM (Wang et al., 2018), which extracts the global semantic coherence of a document via a neural topic model, with the probability of each learned latent topic further adopted to build a mixture-of-experts language model; *7)* TGVAE (Wang et al., 2019b), combining a variational auto-encoder based neural sequence model with a neural topic model; *8)* GBN-RNN, a simplified rGBN-RNN that removes the recurrent structure of its rGBN component; *9)* rGBN-RNN-flipped, which is an additional architectural variation of the proposed rGBN-RNN that modifies $\boldsymbol{\theta}_j^3$ and $\boldsymbol{\theta}_j^1$ shown in Fig. 1(b) by swapping their locations; *10)* Transformer-XL (Dai et al., 2019), which enables learning dependency beyond a fixed length by introducing a recurrence mechanism and a novel position encoding scheme into the Transformer architecture; *11)* GPT-2 (Radford et al., 2019), which can be realized by a generative pre-training of a Transformer-based language model on a diverse set of unlabeled text, followed by discriminative fine-tuning on each specific dataset.

For rGBN-RNN, to ensure the information about the words in the jth sentence to be predicted is not leaking through the sequential document context vectors at the testing stage, the input $\boldsymbol{d}_j$ in (9) only summarizes the preceding sentences $S_{<j}$. For GBN-RNN, following TDLM (Lau et al., 2017) and TCNLM (Wang et al., 2018), all the sentences in a document, excluding the one being predicted, are used to obtain the BoW document context. As shown in Table 1, rGBN-RNN outperforms all RNN-based baselines, and the trend of improvement continues as its number of layers increases, indicating the effectiveness of incorporating recurrent hierarchical topic information into language generation. rGBN-RNN consistently outperforms GBN-RNN, suggesting the benefits of exploiting the sequential dependencies of the sentence-contexts for language modeling.

In Table 1, we further compare the number of parameters between various language models, where we follow the convention to ignore the word embedding layers. The number of parameters for some models are not reported, as we could not find sufficient information from their corresponding papers or code to provide accurate estimations. When used for language generation at the testing stage, rGBN-RNN no longer needs its topics $\{\boldsymbol{\Phi}^l\}$, whose parameters are hence not counted. Note the number of parameters of the topic model component is often dominated by that of the language model component. Table 1 suggests rGBN-RNN, with its hierarchical and temporal topical guidance, achieves better performance with fewer parameters than comparable RNN-based language models.

Note that for language modeling, there has been significant recent interest in replacing RNNs with the Transformer (Vaswani et al., 2017), which consists of stacked multi-head attention modules, and its variants (Dai et al., 2019; Devlin et al., 2019; Radford et al., 2018; 2019). For comparison, we also report the performance of GPT-2 and Transformer-XL, two Transformer-based models. Although shown in Table 1, GPT-2 can obtain better performance than our proposed models, GPT-2 has significantly more parameters and requires a huge text corpus for pre-training. For example, GPT-2 with 12L (Radford et al., 2019) has 117M parameters, while the proposed rGBN-RNN with three hidden layers has as few as 7.3M parameters for language modeling. Moreover, without pre-training, we have tried training the GPT-2 directly with the APNEWS corpus on one machine with 4 NVIDIA RTX 2080 Ti GPUs: even after running 24 hours, the perplexity stays above 600 and does not show a clear trend of improvement as the time progresses. Therefore, we only display in Fig. 2 how Transformer-XL and rGBN-RNN behave during training, by showing the test perplexity of APNEWS documents. It is clear that rGBN-RNN is able to fit the data well, while Transformer-XL behaves well during the early stage of training, it shows a clear trend of overfitting as the training progresses further, possibly because it has an overly large number of model parameters, making it prone to overfitting and hence difficult to generalize.

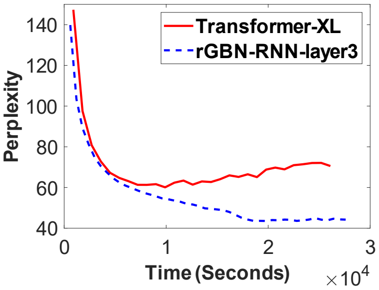


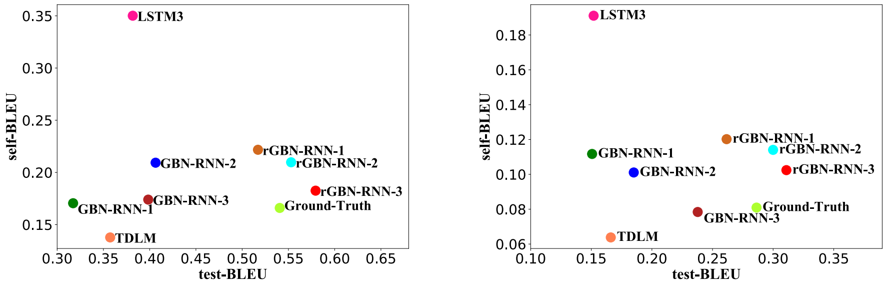
Figure 2. Comparison of Transformer-XL and rGBN-RNN on the test perplexity as a function of training time on APNEWS.

From a structural point of view, we consider the proposed rGBN-RNN as complementary to rather than competing with Transformer based language models, and consider replacing RNN with Transformer to construct a GBN or rGBN guided Transformer as a promising future extension.

BLEU: Following Wang et al. (2019b), we use test-BLEU to evaluate the quality of generated sentences with a set of real test sentences as the reference, and self-BLEU to evaluate the diversity of the generated sentences (Zhu et al., 2018). Given the global parameters of the deep recurrent topic model (rGBN) and language model, we can generate the sentences by following the data generation process of rGBN-RNN: we first generate topic weight vector $\boldsymbol{\theta}_j^L$ randomly and then downward propagate it through rGBN as in (2) to generate $\boldsymbol{\theta}_j^{<L}$. By assimilating the generated topic weight vectors to the hidden states of the language model in each layer, as depicted in (3), we generate a corresponding sentence, where we start from a zero hidden state at each layer in the language model, and sample words sequentially until the end-of-the-sentence symbol is generated. The BLEU scores of various methods are shown in Fig. 3, using the benchmark tool in Texygen (Zhu et al., 2018); We show below BLEU-3 and BLEU-4 for BNC and defer the

Table 1. Comparison of perplexity on three different datasets and the number of parameters when used for language generation.

Model	LSTM Size	#LM Param	Topic Size	#TM Param	#All Param	Perplexity		
						APNEWS	IMDB	BNC
LCLM (Tian & Cho, 2016)	600	—	—	—	—	54.18	67.78	96.50
	900-900	—	—	—	—	50.63	67.86	87.77
LDA+LSTM	600	2.16M	100	0M	2.16M	55.52	69.64	96.50
	900-900	9.72M	100	0M	9.72M	50.75	63.04	87.77
TopicRNN (Dieng et al., 2017)	600	4M	100	4M	4M	54.54	67.83	93.57
	900-900	4M	100	4M	4M	50.24	61.59	84.62
TDLM (Lau et al., 2017)	600	3.33M	100	0.019M	3.35M	52.75	63.45	85.99
	900-900	13.36M	100	0.019M	13.38M	48.97	59.04	81.83
TCNLM (Wang et al., 2018)	600	—	100	—	—	52.63	62.64	86.44
	900-900	—	100	—	—	47.81	56.38	80.14
TGVAE (Wang et al., 2019b)	600	—	50	—	—	48.73	57.11	87.86
basic-LSTM (Hochreiter & Schmidhuber, 1997)	600	2.16M	—	—	2.16M	64.13	72.14	102.89
	900-900	10.80M	—	—	10.80M	58.89	66.47	94.23
	900-900-900	17.28M	—	—	17.28M	60.13	65.16	95.73
GBN-RNN	600	3.4M	100	0.02M	3.42M	47.42	57.01	86.39
	600-512	6.5M	100-80	0.04M	6.54M	44.64	55.42	82.95
	600-512-256	7.2M	100-80-50	0.05M	7.25M	44.35	54.53	80.25
rGBN-RNN	600	3.4M	100	0.03M	3.43M	46.35	55.76	81.94
	600-512	6.5M	100-80	0.06M	6.56M	43.26	53.82	80.25
	600-512-256	7.2M	100-80-50	0.07M	7.27M	**42.71**	**51.36**	**79.13**
rGBN-RNN-flipped	600-512-256	7.2M	100-80-50	0.07M	7.27M	43.55	53.28	81.12
Transformer-XL (Dai et al., 2019)	—	151M	—	—	151M	58.73	60.11	97.14
Pretrained GPT-2 (Radford et al., 2019)	—	117M	—	—	117M	**35.78**	**44.71**	**46.04**



(a) BLEU-3 (b) BLEU-4

Figure 3. BLEU scores of different methods for BNC. BLEU scores towards the lower right corner are preferred.

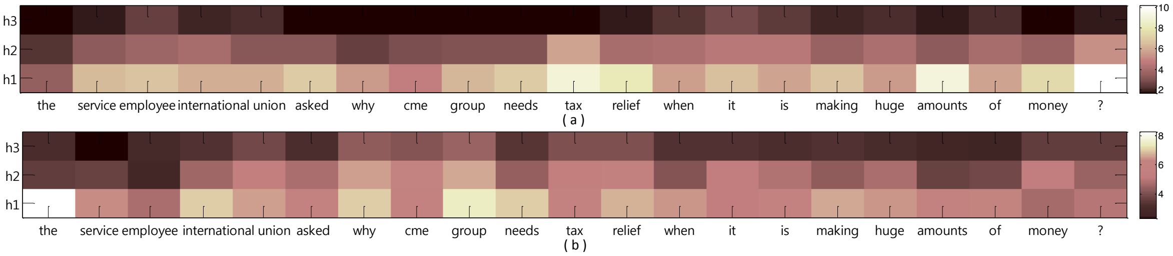
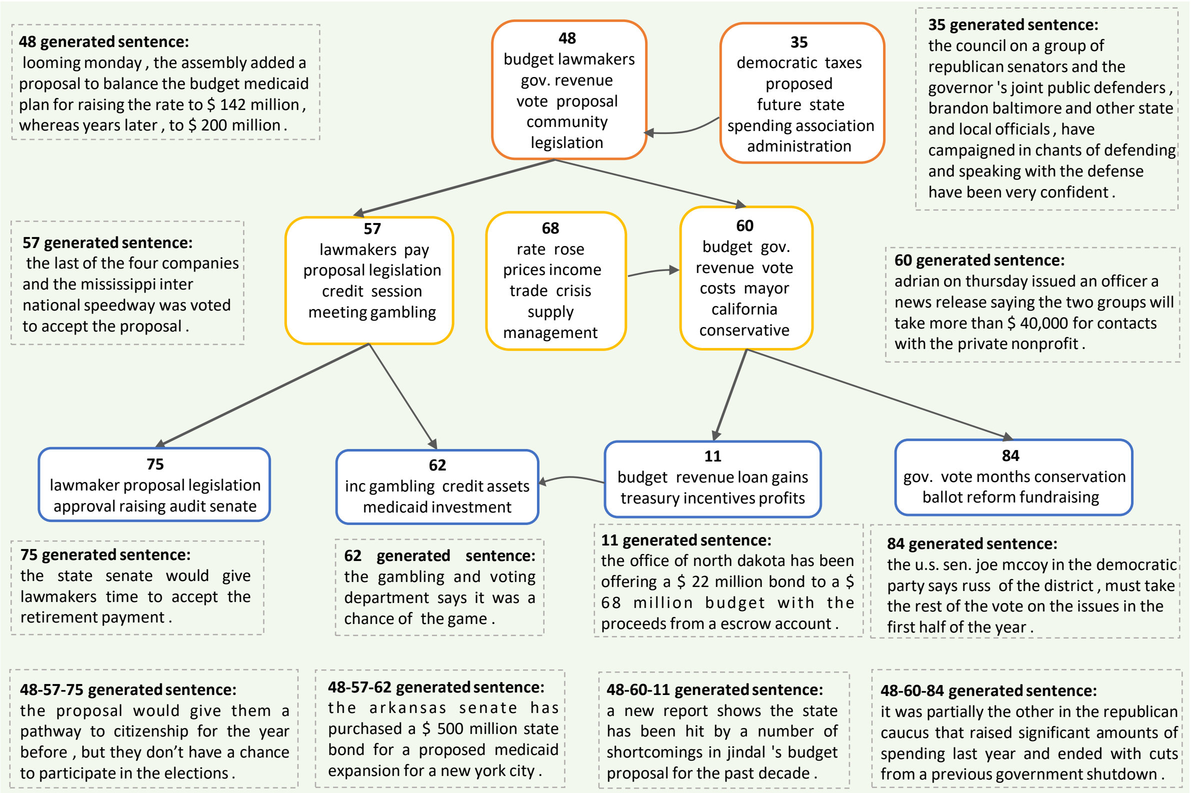


Figure 4. Visualizing the L_2 norms of the hidden states of rGBN-RNN and GBN-RNN, shown in the top and bottom rows, respectively.

analogous plots for IMDB and APNEWS to Appendices E and F, respectively. Note we set the validation dataset as the ground-truth. For all datasets, it is clear that rGBN-RNN yields both higher test-BLEU and lower self-BLEU scores than related methods do, indicating the stacked-RNN based language model in rGBN-RNN generalizes well and does not suffer from mode collapse ($i.e.$, low diversity).

3.2. Qualitative Analysis

Hierarchical structure of language model: In Fig. 4, we visualize the hierarchical multi-scale structures learned with the language model of rGBN-RNN and that of GBN-RNN, by visualizing the L_2-norm of the hidden states in each layer, while reading a sentence from the APNEWS validation set


48
budget lawmakers gov. revenue vote proposal community legislation
35
democratic taxes proposed future state spending association administration
57
lawmakers pay proposal legislation credit session meeting gambling
68
rate rose prices income trade crisis supply management
60
budget gov. revenue vote costs mayor california conservative
75
lawmaker proposal legislation approval raising audit senate
62
inc gambling credit assets medicaid investment
11
budget revenue loan gains treasury incentives profits
84
gov. vote months conservation ballot reform fundraising

48 generated sentence: looming monday , the assembly added a proposal to balance the budget medicaid plan for raising the rate to $ 142 million , whereas years later , to $ 200 million .

35 generated sentence: the council on a group of republican senators and the governor 's joint public defenders , brandon baltimore and other state and local officials , have campaigned in chants of defending and speaking with the defense have been very confident .

57 generated sentence: the last of the four companies and the mississippi inter national speedway was voted to accept the proposal .

60 generated sentence: adrian on thursday issued an officer a news release saying the two groups will take more than $ 40,000 for contacts with the private nonprofit .

75 generated sentence: the state senate would give lawmakers time to accept the retirement payment .

62 generated sentence: the gambling and voting department says it was a chance of the game .

11 generated sentence: the office of north dakota has been offering a $ 22 million bond to a $ 68 million budget with the proceeds from a escrow account .

84 generated sentence: the u.s. sen. joe mccoy in the democratic party says russ of the district , must take the rest of the vote on the issues in the first half of the year .

48-57-75 generated sentence: the proposal would give them a pathway to citizenship for the year before , but they don't have a chance to participate in the elections .

48-57-62 generated sentence: the arkansas senate has purchased a $ 500 million state bond for a proposed medicaid expansion for a new york city .

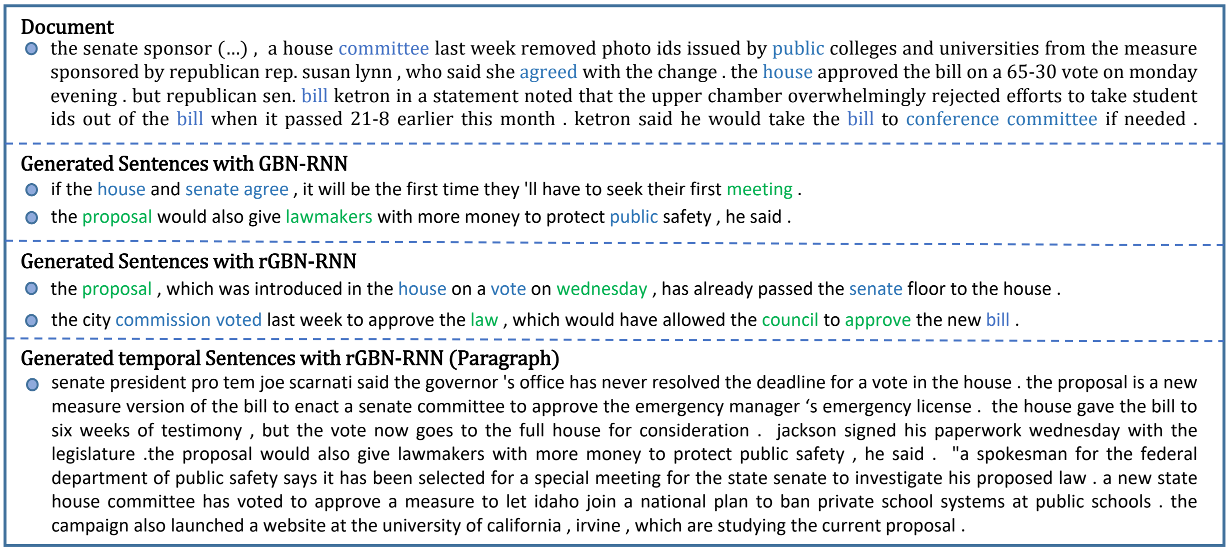
48-60-11 generated sentence: a new report shows the state has been hit by a number of shortcomings in jindal 's budget proposal for the past decade .

48-60-84 generated sentence: it was partially the other in the republican caucus that raised significant amounts of spending last year and ended with cuts from a previous government shutdown .

Figure 5. Example topics and their hierarchical and temporal connections inferred by a three-hidden-layer rGBN-RNN from the APNEWS corpus, and the generated sentences under topic guidance. Top words of each topic at layers 3, 2, and 1 are shown in orange, yellow, and blue boxes, respectively, and each sentence is shown in a dotted line box labeled with the corresponding topic index. Sentences generated with a combination of topics at different layers are shown at the bottom. See the Appendix for analogous plots for both IMDB and BNC.

as "*the service employee international union asked why cme group needs tax relief when it is making huge amounts of money?*" As shown in Fig. 4 (a), in the bottom hidden layer (h1), the L_2 norm sequence varies quickly from word to word, except within short phrases such as "service employee," "international union," and "tax relief," suggesting layer h1 is in charge of capturing short-term local dependencies. By contrast, in the top hidden layer (h3), the L_2 norm sequence varies slowly and exhibits semantic/syntactic meaningful long segments, such as "service employee international union," "asked why cme group needs tax relief," "when it is," and "making huge amounts of," suggesting that layer h3 is in charge of capturing long-range dependencies. Therefore, the language model in rGBN-RNN can allow more specific information to transmit through lower layers, while allowing more general higher level information to transmit through higher layers. Our proposed model have the ability to learn hierarchical structure of the sequence, despite without designing the multiscale RNNs on purpose like Chung et al. (2017). We also visualize the language model of GBN-RNN in Fig. 4 (b); with much less smoothly time-evolved deeper layers, GBN-RNN fails to utilize its stacked RNN structure as effectively as rGBN-RNN does. This suggests that the language model is better trained in rGBN-RNN than in GBN-RNN for capturing long-range temporal dependencies, which helps explain why rGBN-RNN exhibits clearly boosted BLEU scores in comparison to GBN-RNN.

Sentence generation under topic guidance: Given the learned rGBN-RNN, we can sample the sentences both conditioning on a single topic of a certain layer and on a combination of the topics from different layers. Shown in the dotted-line boxes in Fig. 5, most of the generated sentences conditioned on a single topic or a combination of topics are highly related to the given topics in terms of their semantical meanings but not necessarily in key words, indicating the language model is successfully guided by the recurrent hierarchical topics. These observations suggest that rGBN-RNN has successfully captured syntax and global semantics simultaneously for natural language generation. Similar to Fig. 5, we also provide hierarchical topics and corresponding generated sentences for both IMDB and BNC in Appendix G. Besides, in Appendix H, we provide additional example topic hierarchies and generated sentences given different topics.



Document

- the senate sponsor (...) , a house committee last week removed photo ids issued by public colleges and universities from the measure sponsored by republican rep. susan lynn , who said she agreed with the change . the house approved the bill on a 65-30 vote on monday evening . but republican sen. bill ketron in a statement noted that the upper chamber overwhelmingly rejected efforts to take student ids out of the bill when it passed 21-8 earlier this month . ketron said he would take the bill to conference committee if needed .

Generated Sentences with GBN-RNN

- if the house and senate agree , it will be the first time they 'll have to seek their first meeting .
- the proposal would also give lawmakers with more money to protect public safety , he said .

Generated Sentences with rGBN-RNN

- the proposal , which was introduced in the house on a vote on wednesday , has already passed the senate floor to the house .
- the city commission voted last week to approve the law , which would have allowed the council to approve the new bill .

Generated temporal Sentences with rGBN-RNN (Paragraph)

- senate president pro tem joe scarnati said the governor 's office has never resolved the deadline for a vote in the house . the proposal is a new measure version of the bill to enact a senate committee to approve the emergency manager 's emergency license . the house gave the bill to six weeks of testimony , but the vote now goes to the full house for consideration . jackson signed his paperwork wednesday with the legislature .the proposal would also give lawmakers with more money to protect public safety , he said . "a spokesman for the federal department of public safety says it has been selected for a special meeting for the state senate to investigate his proposed law . a new state house committee has voted to approve a measure to let idaho join a national plan to ban private school systems at public schools . the campaign also launched a website at the university of california , irvine , which are studying the current proposal .

Figure 6. An example of generated sentences and paragraph conditioned on a document from APNEWS (green denotes novel words, blue the key words in document and generated sentences.) See the Appendix for analogous plots for both IMDB and BNC.

Hierarchical topics: We present an example topic hierarchy inferred by a three-layer rGBN-RNN from APNEWS. In Fig. 5, we select a large-weighted topic at the top hidden layer and move down the network to include any lower-layer topics connected to their ancestors with sufficiently large weights. Horizontal arrows link temporally related topics at the same layer, while top-down arrows link hierarchically related topics across layers. For example, topic 48 of layer 3 on "budget, lawmakers, gov., revenue" is related not only in hierarchy to topic 57 on "lawmakers, pay, proposal, legislation" and topic 60 of the lower layer on "budget, gov., revenue, vote, costs, mayor," but also in time to topic 35 of the same layer on "democratic, taxes, proposed, future, state." Highly interpretable hierarchical relationships between the topics at different layers, and temporal relationships between the topics at the same layer are captured by rGBN-RNN, and the topics are often quite specific semantically at the bottom layer while becoming increasingly more general when moving upwards.

Sentence/paragraph generation conditioning on a paragraph: Given the GBN-RNN and rGBN-RNN learned on APNEWS, we further present the generated sentences conditioning on a paragraph, as shown in Fig. 6. We provide analogous plots to Fig. 6 for both IMDB and BNC in Appendix I. To randomly generate sentences, we encode the paragraph into a hierarchical latent representation and then feed it into the stacked-RNN. Besides, we can generate a paragraph with rGBN-RNN, using its recurrent inference network to encode the paragraph into a dynamic hierarchical latent representation, which is fed into the language model to predict the word sequence in each sentence of the input paragraph. It is clear that both the proposed GBN-RNN and rGBN-RNN can successfully capture the key textual information of the input paragraph, and generate diverse realistic sentences. Interestingly, the proposed rGBN-RNN can generate semantically coherent paragraphs, incorporating contextual information both within and beyond the sentences. Note that with the topics that extract the document-level word concurrence patterns, our proposed models can generate semantically-related words, which may not exist in the original document.

4. Conclusion

We propose a recurrent gamma belief network (rGBN) guided RNN-based language modeling framework, a novel method to jointly learn a neural language model and a deep recurrent topic model. For scalable inference, we develop hybrid stochastic gradient MCMC and recurrent autoencoding variational inference, allowing efficient end-to-end training. Experiments conducted on real world corpora demonstrate that the proposed models outperform a variety of shallow-topic-model-guided RNN-based language models, and effectively generate the sentences from the designated multi-level topics or noise, while inferring interpretable hierarchical latent topic structures of documents and hierarchical multiscale structures of sequences. For future work, we plan to extend the proposed models to specific natural language processing tasks, such as machine translation, image paragraph captioning, and text summarization. Another promising extension is to replace the stacked-RNN in GBN-RNN or rGBN-RNN with Transformer, $i.e.$, constructing a GBN or rGBN guided Transformer as a new larger-context neural language model.

Acknowledgements

B. Chen acknowledges the support of the Program for Young Thousand Talent by Chinese Central Government, the 111 Project (No. B18039), NSFC (61771361), Shaanxi Innovation Team Project, and the Innovation Fund of Xidian University. M. Zhou acknowledges the support of the U.S. National Science Foundation under Grant IIS-1812699.

References

Acharya, A., Ghosh, J., and Zhou, M. Nonparametric Bayesian factor analysis for dynamic count matrices. In *AISTATS*, 2015.

Ahn, S., Choi, H., Parnamaa, T., and Bengio, Y. A neural knowledge language model. *arXiv: Computation and Language*, 2017.

Blei, D. M. and Lafferty, J. D. Dynamic topic models. In *ICML*, 2006.

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

Cho, K., Merrienboer, B. V., Gulcehre, C., Bougares, F., and Bengio, Y. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *Computer Science*, 2014.

Chung, J., Ahn, S., and Bengio, Y. Hierarchical multiscale recurrent neural networks. In *ICLR*, 2017.

Cong, Y., Chen, B., Liu, H., and Zhou, M. Deep latent Dirichlet allocation with topic-layer-adaptive stochastic gradient Riemannian MCMC. In *ICML*, 2017a.

Cong, Y., Chen, B., and Zhou, M. Fast simulation of hyperplane-truncated multivariate normal distributions. *Bayesian Anal.*, 12(4):1017–1037, 2017b.

Consortium, B. The British National Corpus, version 3 (BNC XML Edition). http://www.natcorp.ox.ac.uk, 2007.

Dai, Z., Yang, Z., Yang, Y., Carbonell, J. G., Le, Q. V., and Salakhutdinov, R. Transformer-xl: Attentive language models beyond a fixed-length context. In *ACL*, 2019.

Devlin, J., Chang, M., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. In *north american chapter of the association for computational linguistics*, pp. 4171–4186, 2019.

Dieng, A. B., Wang, C., Gao, J., and Paisley, J. TopicRNN: A recurrent neural network with long-range semantic dependency. In *ICLR*, 2017.

Fan, X., Zhang, Y., Wang, Z., and Zhou, M. Adaptive correlated Monte Carlo for contextual categorical sequence generation. In *International Conference on Learning Representations*, 2020.

Gan, Z., Chen, C., Henao, R., Carlson, D., and Carin, L. Scalable deep Poisson factor analysis for topic modeling. In *ICML*, pp. 1823–1832, 2015.

Gan, Z., Gan, C., He, X., Pu, Y., Tran, K., Gao, J., Carin, L., and Deng, L. Semantic compositional networks for visual captioning. In *CVPR*, pp. 1141–1150, 2017.

Gehrmann, S., Deng, Y., and Rush, A. Bottom-up abstractive summarization. In *EMNLP*, pp. 4098–4109, 2018.

Girolami, M. A. and Calderhead, B. Riemann manifold Langevin and Hamiltonian Monte Carlo methods. *Journal of The Royal Statistical Society Series B-statistical Methodology*, 73(2):123–214, 2011.

Graves, A. Generating sequences with recurrent neural networks. *arXiv: Neural and Evolutionary Computing*, 2013.

Graves, A., Mohamed, A., and Hinton, G. Speech recognition with deep recurrent neural networks. In *ICASSP*, pp. 6645–6649, 2013.

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

Griffiths, T. L., Steyvers, M., Blei, D. M., and Tenenbaum, J. B. Integrating topics and syntax. In *NeurIPS*, pp. 537–544, 2004.

Guo, D., Chen, B., Zhang, H., and Zhou, M. Deep Poisson gamma dynamical systems. In *NeurIPS*, pp. 8451–8461, 2018.

Hochreiter, S. and Schmidhuber, J. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

Hoffman, M. D., Blei, D. M., Wang, C., and Paisley, J. Stochastic variational inference. *The Journal of Machine Learning Research*, 14(1):1303–1347, 2013.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. In *ICLR*, 2015.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. In *ICLR*, 2013.

Klein, D. and Manning, C. D. Accurate unlexicalized parsing. In *Meeting of the Association for Computational Linguistics*, 2003.

Lau, J. H., Baldwin, T., and Cohn, T. Topically driven neural language model. In *meeting of the association for computational linguistics*, pp. 355–365, 2017.

Li, C., Chen, C., Carlson, D., and Carin, L. Preconditioned stochastic gradient Langevin dynamics for deep neural networks. *arXiv*, 2015.

Ma, Y., Chen, T., and Fox, E. A complete recipe for stochastic gradient MCMC. In *NIPS*, pp. 2899–2907, 2015.

Maas, A. L., Daly, R. E., Pham, P. T., Dan, H., Ng, A. Y., and Potts, C. Learning Word Vectors for Sentiment Analysis. In *Meeting of the Association for Computational Linguistics: Human Language Technologies*, 2011.

Mao, J., Xu, W., Yang, Y., Wang, J., Huang, Z., and Yuille, A. L. Deep captioning with multimodal recurrent neural networks m-RNN. In *ICLR*, 2015.

Miao, Y., Grefenstette, E., and Blunsom, P. Discovering discrete latent topics with neural variational inference. In *ICML*, pp. 2410–2419, 2017.

Mikolov, T. and Zweig, G. Context dependent recurrent neural network language model. In *SLT*, pp. 234–239, 2012.

Mikolov, T., Karafiat, M., Burget, L., Cernocky, J., and Khudanpur, S. Recurrent neural network based language model. In *Interspeech*, 2010.

Mikolov, T., Kombrink, S., Burget, L., Cernocky, J., and Khudanpur, S. Extensions of recurrent neural network language model. In *ICASSP*, pp. 5528–5531, 2011.

Mnih, A. and Gregor, K. Neural variational inference and learning in belief networks. In *ICML*, pp. 1791–1799, 2014.

Patterson, S. and Teh, Y. W. Stochastic gradient Riemannian Langevin dynamics on the probability simplex. In *NIPS*, pp. 3102–3110, 2013.

Radford, A., Narasimhan, K., Salimans, T., and Sutskever, I. Improving language understanding by generative pre-training. 2018.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. Language models are unsupervised multitask learners. 2019.

Rennie, S. J., Marcheret, E., Mroueh, Y., Ross, J., and Goel, V. Self-critical sequence training for image captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 7008–7024, 2017.

Rezende, D. J., Mohamed, S., and Wierstra, D. Stochastic backpropagation and approximate inference in deep generative models. In *ICML*, pp. 1278–1286, 2014.

Rush, A. M., Chopra, S., and Weston, J. A neural attention model for abstractive sentence summarization. In *EMNLP*, pp. 379–389, 2015.

Schein, A., Wallach, H., and Zhou, M. Poisson–gamma dynamical systems. In *Neural Information Processing Systems*, 2016.

Srivastava, A. and Sutton, C. Autoencoding variational inference for topic models. In *ICLR*, 2017.

Srivastava, N., Salakhutdinov, R., and Hinton, G. E. Modeling documents with deep Boltzmann machines. In *Uncertainty in Artificial Intelligence*, 2013.

Sutskever, I., Vinyals, O., and Le, Q. V. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*, pp. 3104–3112, 2014.

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

Tian, W. and Cho, K. Larger-context language modelling with recurrent neural network. In *Meeting of the Association for Computational Linguistics*, 2016.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., and Polosukhin, I. Attention is all you need. In *Neural Information Processing Systems*, pp. 6000–6010, 2017.

Vinyals, O., Toshev, A., Bengio, S., and Erhan, D. Show and tell: A neural image caption generator. In *CVPR*, 2015.

Wallach, H. M. Topic modeling: beyond bag-of-words. In *ICML*, pp. 977–984, 2006.

Wang, C., Chen, B., Xiao, S., and Zhou, M. Convolutional Poisson gamma belief network. In *International Conference on Machine Learning*, pp. 6515–6525, 2019a.

Wang, W., Gan, Z., Wang, W., Shen, D., Huang, J., Ping, W., Satheesh, S., and Carin, L. Topic compositional neural language model. In *AISTATS*, pp. 356–365, 2018.

Wang, W., Gan, Z., Xu, H., Zhang, R., Wang, G., Shen, D., Chen, C., and Carin, L. Topic-guided variational autoencoders for text generation. In *NAACL*, 2019b.

Welling, M. and Teh, Y. W. Bayesian learning via stochastic gradient Langevin dynamics. In *ICML*, pp. 681–688, 2011.

Xu, K., Ba, J., Kiros, R., Cho, K., Courville, A. C., Salakhutdinov, R., Zemel, R. S., and Bengio, Y. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*, 2015.

Zhang, H., Chen, B., Guo, D., and Zhou, M. WHAI: Weibull hybrid autoencoding inference for deep topic modeling. In *ICLR*, 2018.

Zhao, H., Du, L., Buntine, W., and Zhou, M. Dirichlet belief networks for topic structure learning. In *Neural Information Processing Systems*, pp. 7955–7966, 2018.

Zhou, M. and Carin, L. Negative binomial process count and mixture modeling. *IEEE Trans. Pattern Analysis and Machine Intelligence*, 37(2):307–320, 2015.

Zhou, M., Hannah, L., Dunson, D. B., and Carin, L. Beta-negative binomial process and Poisson factor analysis. In *AISTATS*, pp. 1462–1471, 2012.

Zhou, M., Cong, Y., and Chen, B. The Poisson gamma belief network. In *NIPS*, pp. 3025–3033, 2015.

Zhou, M., Cong, Y., and Chen, B. Augmentable gamma belief networks. *J. Mach. Learn. Res.*, 17(163):1–44, 2016.

Zhu, Y., Lu, S., Zheng, L., Guo, J., Zhang, W., Wang, J., and Yu, Y. Texygen: A benchmarking platform for text generation models. *SIGIR*, 2018.