Topic-Guided Variational Autoencoders for Text Generation

Wenlin Wang[1], **Zhe Gan**[2], **Hongteng Xu**[1,3], **Ruiyi Zhang**[1], **Guoyin Wang**[1], **Dinghan Shen**[1], **Changyou Chen**[4], **Lawrence Carin**[1]
[1]Duke University, [2]Microsoft Dynamics 365 AI Research,
[3]Infinia ML, Inc, [4]University at Buffalo
{wenlin.wang, ruiyi.zhang, gw60, ds448, lcarin}@duke.edu
zhe.gan@microsoft.com, hongteng.xu@infiniaml.com
changyou@buffalo.edu

Abstract

We propose a topic-guided variational autoencoder (TGVAE) model for text generation. Distinct from existing variational autoencoder (VAE) based approaches, which assume a simple Gaussian prior for the latent code, our model specifies the prior as a Gaussian mixture model (GMM) parametrized by a neural topic module. Each mixture component corresponds to a latent topic, which provides guidance to generate sentences under the topic. The neural topic module and the VAE-based neural sequence module in our model are learned jointly. In particular, a sequence of invertible Householder transformations is applied to endow the approximate posterior of the latent code with high flexibility during model inference. Experimental results show that our TGVAE outperforms alternative approaches on both unconditional and conditional text generation, which can generate semantically-meaningful sentences with various topics.

1 Introduction

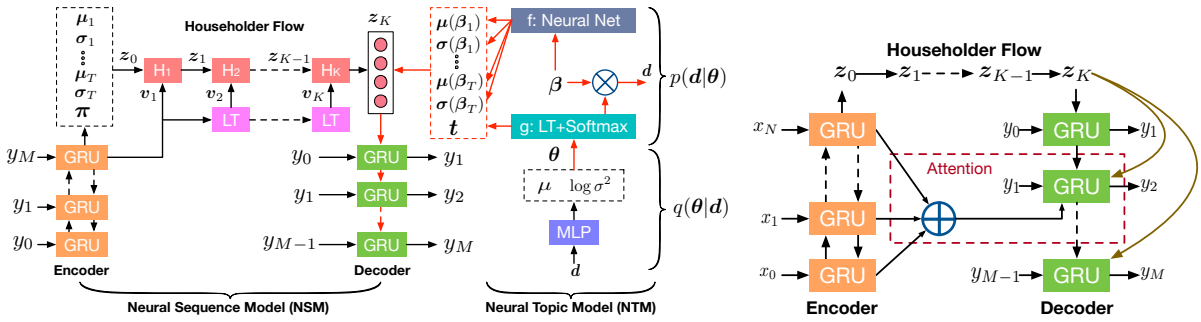
Text generation plays an important role in various natural language processing (NLP) applications, such as machine translation (Cho et al., 2014; Sutskever et al., 2014), dialogue generation (Li et al., 2017a), and text summarization (Nallapati et al., 2016; Rush et al., 2015). As a competitive solution to this task, the variational autoencoder (VAE) (Kingma and Welling, 2013; Rezende et al., 2014) has been widely used in text-generation systems (Bowman et al., 2015; Hu et al., 2017; Serban et al., 2017). In particular, VAE defines a generative model that propagates latent codes drawn from a simple prior through a decoder to manifest data samples. The generative model is further augmented with an inference network, that feeds observed data samples through an encoder to yield a distribution on the corresponding latent codes.

Compared with other potential methods, $e.g.$, those based on generative adversarial networks (GANs) (Yu et al., 2017; Guo et al., 2017; Zhang et al., 2017b, 2018; Chen et al., 2018), VAE is of particular interest when one desires not only text generation, but also the capacity to infer meaningful latent codes from text. Ideally, semantically-meaningful latent codes can provide high-level guidance while generating sentences. For example, when generating text, the vocabulary could potentially be narrowed down if the input latent code corresponds to a certain topic (*e.g.*, the word "military" is unlikely to appear in a sports-related document).

However, in practice this desirable property is not fully achieved by existing VAE-based text generative models, because of the following two challenges. First, the sentences in documents may associate with different semantic information (*e.g.*, topic, sentiment, etc.) while the latent codes of existing VAE-based text generative models often employ a simple Gaussian prior, which cannot indicate the semantic structure among sentences and may reduce the generative power of the decoder. Although some variants of VAE try to impose some structure on the latent codes (Jiang et al., 2016; Dilokthanakul et al., 2016), they are often designed with pre-defined parameter settings without incorporating semantic meanings into the latent codes, which may lead to over-regularization (Dilokthanakul et al., 2016).

The second issue associated with VAE-based text generation is "posterior collapse," first identified in Bowman et al. (2015). With a strong auto-regressive decoder network (*e.g.*, LSTM), the model tends to ignore the information from the latent code and merely depends on previous generated tokens for prediction. Several strategies are proposed to mitigate this problem, including making the decoder network less auto-regressive ($i.e.$,

Proceedings of NAACL-HLT 2019, pages 166–177
Minneapolis, Minnesota, June 2 - June 7, 2019.



(a) The scheme of our method.

(b) The extension to text summarization.

Figure 1: Illustration of the proposed Topic-Guided Variational Autoencoder (TGVAE) for text generation. (a) For generation (the red arrows), the topics inferred from a neural topic model are used to guide a Gaussian mixture prior of the latent code, which is further fed into the decoder to generate a sentence. For inference (the black arrows), the sentence is encoded into a vector and then propagated through the Householder flow to obtain the approximate posterior. (b) An attention module is further added for text summarization. The same neural topic model is also applied, but omitted here for simplicity of illustration. "LT" denotes a linear transformation.

using less conditional information while generating each word) (Yang et al., 2017; Shen et al., 2017a), or bridging the amortization gap (between the log-likelihood and the ELBO) using semi-amortized inference networks (Kim et al., 2018). However, these methods mitigate the issue by weakening the conditional dependency on the decoder, which may fail to generate high-quality continuous sentences.

To overcome the two problems mentioned above, we propose a topic-guided variational autoencoder (TGVAE) model, permitting text generation with designated topic guidance. As illustrated in Figure 1(a), TGVAE specifies a Gaussian mixture model (GMM) as the prior of the latent code, where each mixture component corresponds to a topic. The GMM is learnable based on a neural topic model — the mean and diagonal covariance of each mixture component is parameterized by the corresponding topic. Accordingly, the degree to which each component of the GMM is used to generate the latent code and the corresponding sentence is tied to the usage of the topics. In the inference phase, we initialize the latent code from a GMM generated via the encoder, and apply the invertible Householder transformation (Bischof and Sun, 1994; Sun and Bischof, 1995) to derive the latent code with high flexibility and low complexity.

As shown in Figure 1(b), besides unconditional text generation, the proposed model can be extended for conditional text generation, *i.e.,* abstractive text summarization (Nallapati et al., 2016) with an attention module. By injecting the topics learned by our model (semantic information), we are able to make better use of the source document and improve a sequence-to-sequence summarization model (Sutskever et al., 2014).

We highlight the contributions of our model as follows: (*i*) A new Topic-Guided VAE (TGVAE) model is proposed for text generation with designated topic guidance. (*ii*) For the model inference, Householder flow is introduced to transform a relatively simple mixture distribution into an arbitrarily flexible approximate posterior, achieving powerful approximate posterior inference. (*iii*) Experiments for both unconditional and conditional text generation demonstrate the effectiveness of the proposed approach.

2 Model

The proposed TGVAE, as illustrated in Figure 1(a), consists of two modules: a neural topic model (NTM) and a neural sequence model (NSM). The NTM aims to capture long-range semantic meaning across the document, while the NSM is designed to generate a sentence with designated topic guidance.

2.1 Neural Topic Model

Let $\boldsymbol{d} \in \mathbb{Z}_+^D$ denote the bag-of-words representation of a document, with $\mathbb{Z}_+$ denoting non-negative integers. D is the vocabulary size, and each element of $\boldsymbol{d}$ reflects a count of the number of times the corresponding word occurs in the document. Let a_n represent the topic assignment for word w_n. Following Miao et al. (2017), a Gaussian random vector is passed through a softmax function to parameterize the multinomial document topic distributions. Specifically, the generative process

of the NTM is

$$\begin{aligned} \boldsymbol{\theta} \sim \mathcal{N}(0, \mathbf{I}), \quad \boldsymbol{t} = g(\boldsymbol{\theta}), \\ a_n \sim \text{Discrete}(\boldsymbol{t}), \quad w_n \sim \text{Discrete}(\boldsymbol{\beta}_{a_n}), \end{aligned} \tag{1}$$

where $\mathcal{N}(0, \mathbf{I})$ is an isotropic Gaussian distribution, $g(\cdot)$ is a transformation function that maps sample $\boldsymbol{\theta}$ to the topic embedding $\boldsymbol{t}$, defined here as $g(\boldsymbol{\theta}) = \text{softmax}(\hat{\mathbf{W}}\boldsymbol{\theta} + \hat{\boldsymbol{b}})$, where $\hat{\mathbf{W}}$ and $\hat{\boldsymbol{b}}$ are trainable parameters; $\boldsymbol{\beta}_{a_n}$ represents the distribution over words for topic a_n; $n \in [1, N_d]$, and N_d is the number of words in the document. The marginal likelihood for document $\boldsymbol{d}$ is:

$$\begin{aligned} p(\boldsymbol{d}|\boldsymbol{\beta}) &= \int_{\boldsymbol{t}} p(\boldsymbol{t}) \prod\nolimits_n \sum\nolimits_{a_n} p(w_n|\boldsymbol{\beta}_{a_n}) p(a_n|\boldsymbol{t}) d\boldsymbol{t} \\ &= \int_{\boldsymbol{t}} p(\boldsymbol{t}) \prod\nolimits_n p(w_n|\boldsymbol{\beta}, \boldsymbol{t}) d\boldsymbol{t} \\ &= \int_{\boldsymbol{t}} p(\boldsymbol{t}) p(\boldsymbol{d}|\boldsymbol{\beta}, \boldsymbol{t}) d\boldsymbol{t} = \int_{\boldsymbol{\theta}} p(\boldsymbol{\theta}) p(\boldsymbol{d}|\boldsymbol{\beta}, \boldsymbol{\theta}) d\boldsymbol{\theta}. \end{aligned} \tag{2}$$

The second equation in (2) holds because we can marginalize out the sampled topic words a_n by

$$p(w_n|\boldsymbol{\beta}, \boldsymbol{t}) = \sum\nolimits_{a_n} p(w_n|\boldsymbol{\beta}_{a_n}) p(a_n|\boldsymbol{t}) = \boldsymbol{\beta}\boldsymbol{t}, \tag{3}$$

where $\boldsymbol{\beta} = \{\boldsymbol{\beta}_i\}_{i=1}^T$ are trainable parameters of the decoder; T is the number of topics and each $\boldsymbol{\beta}_i \in \mathbb{R}^D$ is a topic distribution over words (all elements of $\boldsymbol{\beta}_i$ are nonnegative, and sum to one).

2.2 Neural Sequence Model

Our neural sequence model for text generation is built upon the VAE proposed in Bowman et al. (2015). Specifically, a continuous latent code $\boldsymbol{z}$ is first generated from some prior distribution $p(\boldsymbol{z})$, based on which the text sequence $\boldsymbol{y}$ is then generated from a conditional distribution $p(\boldsymbol{y}|\boldsymbol{z})$ parameterized by a neural network (often called the decoder). Since the model incorporates a latent variable $\boldsymbol{z}$ that modulates the entire generation of the sentence, it should be able to capture the high-level source of variation in the data.

Topic-Guided Gaussian Mixture Prior The aforementioned intuition is hard to be captured by a standard VAE, simply imposing a Gaussian prior on top of $\boldsymbol{z}$, since the semantic information associated with a document intrinsically contains different subgroups (such as topics, sentiment, etc.). In our model, we consider incorporating the topic information into latent variables. Our model assumes each $\boldsymbol{z}$ is drawn from a *topic-dependent* GMM, that is,

$$\begin{aligned} p(\boldsymbol{z}|\boldsymbol{\beta}, \boldsymbol{t}) &= \sum\nolimits_{i=1}^T t_i \mathcal{N}(\boldsymbol{\mu}(\boldsymbol{\beta}_i), \boldsymbol{\sigma}^2(\boldsymbol{\beta}_i)) \\ \boldsymbol{\mu}(\boldsymbol{\beta}_i) &= f_\mu(\boldsymbol{\beta}_i) \\ \boldsymbol{\sigma}^2(\boldsymbol{\beta}_i) &= \text{diag}(\exp(f_\sigma(\boldsymbol{\beta}_i))), \end{aligned} \tag{4}$$

where t_i is the usage of topic i in a document and $\boldsymbol{\beta}_i$ is the i-th topic distribution over words. Both of them are inherited from the NTM discussed above. Both $f_\mu(\cdot)$ and $f_\sigma(\cdot)$ are implemented as feedforward neural networks, with trainable parameters $\mathbf{W}_\mu$ and $\mathbf{W}_\sigma$, respectively. Compared with a normal GMM prior that sets each mixture component to be $\mathcal{N}(0, \mathbf{I})$, the proposed topic guided GMM prior provides semantic meaning for each mixture component, and hence makes the model more interpretable and controllable for text generation.

Decoder The likelihood of a word sequence $\boldsymbol{y} = \{y_m\}_{m=1}^M$ conditioned on the latent code $\boldsymbol{z}$ is defined as:

$$\begin{aligned} p(\boldsymbol{y}|\boldsymbol{z}) &= p(y_1|\boldsymbol{z}) \prod\nolimits_{m=2}^M p(y_m|y_{1:m-1}, \boldsymbol{z}) \\ &= p(y_1|\boldsymbol{z}) \prod\nolimits_{m=2}^M p(y_m|\boldsymbol{h}_m), \end{aligned} \tag{5}$$

where the conditional probability of each word y_m given all the previous words $y_{1:m-1}$ and the latent code $\boldsymbol{z}$ is defined through the hidden state $\boldsymbol{h}_m$: $\boldsymbol{h}_m = f(\boldsymbol{h}_{m-1}, y_{m-1}, \boldsymbol{z})$, where the function $f(\cdot)$ is implemented as a Gated Recurrent Unit (GRU) cell (Cho et al., 2014) in our experiments.

3 Inference

The proposed model (see Figure 1(a)) takes the bag-of-words as input and embeds a document into a topic vector. The topic vector is then used to reconstruct the bag-of-words input, and the learned topic distribution over words is used to model a topic-dependent prior to generate a sentence in the VAE setup. Specifically, the joint marginal likelihood can be written as:

$$\begin{aligned} p(\boldsymbol{y}, \boldsymbol{d}|\boldsymbol{\beta}) = \int_{\boldsymbol{\theta}} \int_{\boldsymbol{z}} p(\boldsymbol{\theta}) p(\boldsymbol{d}|\boldsymbol{\beta}, \boldsymbol{\theta}) \\ \cdot p(\boldsymbol{z}|\boldsymbol{\beta}, \boldsymbol{\theta}) p(\boldsymbol{y}|\boldsymbol{z}) \, d\boldsymbol{\theta} d\boldsymbol{z}. \end{aligned} \tag{6}$$

Since direct optimization of (6) is intractable, auto-encoding variational Bayes is employed (Kingma and Welling, 2013). Denote $q(\boldsymbol{\theta}|\boldsymbol{d})$ and $q(\boldsymbol{z}|\boldsymbol{y})$ as the variational distributions for $\boldsymbol{\theta}$ and $\boldsymbol{z}$, respectively. The variational objective function, also

called the evidence lower bound (ELBO), is constructed as

$$\mathcal{L} = \underbrace{\mathbb{E}_{q(\boldsymbol{\theta}|\boldsymbol{d})}\left[\log p(\boldsymbol{d}|\boldsymbol{\beta},\boldsymbol{\theta})\right] - \mathrm{KL}\left(q(\boldsymbol{\theta}|\boldsymbol{d})||p(\boldsymbol{\theta})\right)}_{\text{neural topic model}, \mathcal{L}_t} + \tag{7}$$
$$\underbrace{\mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{y})}\left[\log p(\boldsymbol{y}|\boldsymbol{z})\right] - \mathbb{E}_{q(\boldsymbol{\theta}|\boldsymbol{d})}\left[\mathrm{KL}\left(q(\boldsymbol{z}|\boldsymbol{y})||p(\boldsymbol{z}|\boldsymbol{\beta},\boldsymbol{\theta})\right)\right]}_{\text{neural sequence model}, \mathcal{L}_s} .$$

By assuming

$$q(\boldsymbol{\theta}|\boldsymbol{d}) = \mathcal{N}(\boldsymbol{\theta}|g_\mu(\boldsymbol{d}), \mathrm{diag}(\exp\left(g_\sigma(\boldsymbol{d})\right))),$$

where both $g_\mu(\cdot)$ and $g_\sigma(\cdot)$ are implemented as feed-forward neural networks, the reparameterization trick (Kingma and Welling, 2013) can be applied directly to build an unbiased and low-variance gradient estimator for the $\mathcal{L}_t$ term in (7). Below, we discuss in detail how to approximate the $\mathcal{L}_s$ term in (7) and infer an arbitrarily complex posterior for $\boldsymbol{z}$. Note that $\boldsymbol{z}$ is henceforth represented as $\boldsymbol{z}_K$ in preparation for the introduction of Householder flows.

3.1 Householder Flow for Approximate Posterior

Householder flow (Zhang et al., 2017a; Tomczak and Welling, 2016) is a volume-preserving normalizing flow (Rezende and Mohamed, 2015), capable of constructing an arbitrarily complex posterior $q_K(\boldsymbol{z}_K|\boldsymbol{y})$ from an initial random variable $\boldsymbol{z}_0$ with distribution q_0, by composing a sequence of invertible mappings, *i.e.*, $\boldsymbol{z}_K = f_K \circ \cdots \circ f_2 \circ f_1(\boldsymbol{z}_0)$. By repeatedly applying the chain rule and using the property of Jacobians of invertible functions, $q_K(\boldsymbol{z}_K|\boldsymbol{y})$ is expressed as:

$$\log q_K(\boldsymbol{z}_K|\boldsymbol{y}) = \log q_0(\boldsymbol{z}_0|\boldsymbol{y}) - \sum_{k=1}^{K} \log \left| \det \frac{\partial f_k}{\partial \boldsymbol{z}_{k-1}} \right|, \tag{8}$$

where $|\det \frac{\partial f_k}{\partial \boldsymbol{z}_{k-1}}|$ is the absolute value of the Jacobian determinant. Therefore, the $\mathcal{L}_s$ term in (7) may be rewritten as

$$\mathbb{E}_{q_0(\boldsymbol{z}_0|\boldsymbol{y})}[\log p(\boldsymbol{y}|\boldsymbol{z}_K)] + \sum_{k=1}^{K} \log \left| \det \frac{\partial f_k}{\partial \boldsymbol{z}_{k-1}} \right|$$
$$-\mathbb{E}_{q(\boldsymbol{\theta}|\boldsymbol{d})}\left[\mathrm{KL}(q_0(\boldsymbol{z}_0|\boldsymbol{y})||p(\boldsymbol{z}_K|\boldsymbol{\beta},\boldsymbol{\theta}))\right] . \tag{9}$$

Here $q_0(\boldsymbol{z}_0|\boldsymbol{y})$ is also specified as a GMM, *i.e.*, $q_0(\boldsymbol{z}_0|\boldsymbol{y}) = \sum_{i=1}^{T} \pi_i(\boldsymbol{y})\mathcal{N}(\boldsymbol{\mu}_i(\boldsymbol{y}), \boldsymbol{\sigma}_i^2(\boldsymbol{y}))$. As illustrated in Figure 1(a), $\boldsymbol{y}$ is first represented as a hidden vector $\boldsymbol{h}$, by encoding the text sequence with an RNN. Based on this, the mixture probabilities $\boldsymbol{\pi}$, the means and diagonal covariances of all the mixture components are all produced by an encoder network, which is a linear layer with the input $\boldsymbol{h}$. In (9), the first term can be considered as the reconstruction error, while the remaining two terms act as regularizers, the tractability of which is important for the whole framework.

KL Divergence between two GMMs Since both the prior $p(\boldsymbol{z}_K|\boldsymbol{\beta},\boldsymbol{\theta})$ and the initial density $q_0(\boldsymbol{z}_0|\boldsymbol{y})$ for the posterior are GMMs, the calculation of the third term in (9) requires the KL divergence between two GMMs. Though no closed-form solutions exist, the KL divergence has an explicit upper bound (Dilokthanakul et al., 2016), shown in Proposition 1.

Proposition 1. *For any two mixture densities* $p = \sum_{i=1}^{n} \pi_i g_i$ *and* $\hat{p} = \sum_{i=1}^{n} \hat{\pi}_i \hat{g}_i$, *their KL divergence is upper-bounded by*

$$\mathrm{KL}\left(p||\hat{p}\right) \leq \mathrm{KL}\left(\pi||\hat{\pi}\right) + \sum_{i=1}^{n} \pi_i \mathrm{KL}\left(g_i||\hat{g}_i\right) , \tag{10}$$

where equality holds if and only if $\frac{\pi_i g_i}{\sum_{i=1}^{n} \pi_i g_i} = \frac{\hat{\pi}\hat{g}_i}{\sum_{i=1}^{n} \hat{\pi}\hat{g}_i}$.

Proof. With the log-sum inequality

$$\begin{aligned}
\mathrm{KL}\left(p||\hat{p}\right) &= \int \left(\sum_i \pi_i g_i\right) \log \frac{\sum_i \pi_i g_i}{\sum_i \hat{\pi}\hat{g}_i} \\
&\leq \int \sum_i \pi_i g_i \log \frac{\pi_i g_i}{\hat{\pi}\hat{g}_i} \\
&= \sum_i \pi_i \log \frac{\pi_i}{\hat{\pi}} + \sum_i \pi_i \int g_i \log \frac{g_i}{\hat{g}_i} \\
&= \mathrm{KL}(\pi||\hat{\pi}) + \sum_i \pi_i \mathrm{KL}(g_i||\hat{g}_i) .
\end{aligned} \tag{11}$$

Since the KL divergence between two Gaussian distributions has a closed-form expression, the upper bound of the KL divergence between two GMMs can be readily calculated. Accordingly, the third term in (9) is upper bounded as

$$\mathcal{U}_{KL} = \mathbb{E}_{q(\boldsymbol{\theta}|\boldsymbol{d})}\Big[\mathrm{KL}\left(\boldsymbol{\pi}(\boldsymbol{y})||\boldsymbol{t}\right) \tag{12}$$
$$+ \sum_{i=1}^{T} \pi_i(\boldsymbol{y}) \mathrm{KL}\left(\mathcal{N}(\boldsymbol{\mu}_i(\boldsymbol{y}), \boldsymbol{\sigma}_i^2(\boldsymbol{y}))||\mathcal{N}(\boldsymbol{\mu}(\boldsymbol{\beta}_i), \boldsymbol{\sigma}^2(\boldsymbol{\beta}_i))\right)\Big],$$

where the expectation $\mathbb{E}_{q(\boldsymbol{\theta}|\boldsymbol{d})}[\cdot]$ can be approximated by a sample from $q(\boldsymbol{\theta}|\boldsymbol{d})$.

Householder Flow Householder flow (Tomczak and Welling, 2016) is a series of Householder transformations, defined as follows. For a given vector $\boldsymbol{z}_{k-1}$, the reflection hyperplane can be defined by a Householder vector $\boldsymbol{v}_t$ that is orthogonal to the hyperplane. The reflection of this point about the hyperplane is

$$\boldsymbol{z}_k = \left(\mathbf{I} - 2\frac{\boldsymbol{v}_k \boldsymbol{v}_k^T}{||\boldsymbol{v}_k||^2}\right) \boldsymbol{z}_{k-1} = \mathbf{H}_k \boldsymbol{z}_{k-1} , \tag{13}$$

where $\mathbf{H}_k = \mathbf{I} - 2\frac{\boldsymbol{v}_k \boldsymbol{v}_k^T}{||\boldsymbol{v}_k||^2}$ is called the *Householder matrix*. An important property of the *Householder matrix* is that the absolute value of the Jacobian determinant is equal to 1, therefore $\sum_{k=1}^{K} \log \left| \det \frac{\partial f_k}{\partial \boldsymbol{z}_{k-1}} \right| = \sum_{k=1}^{K} \log |\det \mathbf{H}_k| = 0$, significantly simplifying the computation of the lower bound in (9). For $k = 1, \ldots, K$, the vector $\boldsymbol{v}_k$ is produced by a linear layer with the input $\boldsymbol{v}_{k-1}$, where $\boldsymbol{v}_0 = \boldsymbol{h}$ is the last hidden vector of the encoder RNN that encodes the sentence $\boldsymbol{y}$.

Finally, by combining (7), (9) and (12), the ELBO can be rewritten as

$$\mathcal{L} \geq \mathcal{L}_t + \mathbb{E}_{q_0(\boldsymbol{z}_0|\boldsymbol{y})}[\log p(\boldsymbol{y}|\boldsymbol{z}_K)] - \mathcal{U}_{KL} \,. \quad (14)$$

3.2 Extension to text summarization

When extending our model to text summarization, we are interested in modeling $p(\boldsymbol{y}, \boldsymbol{d}|\boldsymbol{x})$, where $(\boldsymbol{x}, \boldsymbol{y})$ denotes the document-summary pair, and $\boldsymbol{d}$ denotes the bag-of-words of the input document. The marginal likelihood can be written as $p(\boldsymbol{y}, \boldsymbol{d}|\boldsymbol{x}) = \int_{\boldsymbol{\theta}} \int_{\boldsymbol{z}} p(\boldsymbol{\theta}) p(\boldsymbol{d}|\boldsymbol{\beta}, \boldsymbol{\theta}) p(\boldsymbol{z}|\boldsymbol{\beta}, \boldsymbol{\theta}) p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{z}) \, d\boldsymbol{\theta} d\boldsymbol{z}$. Assume the approximate posterior of $\boldsymbol{z}$ is only dependent on $\boldsymbol{x}$, *i.e.*, $q(\boldsymbol{z}|\boldsymbol{x})$ is proposed as the variational distribution for $\boldsymbol{z}$. The ELBO is then constructed as

$$\begin{aligned} \mathcal{L} =& \mathcal{L}_t + \mathbb{E}_{q(\boldsymbol{z}|\boldsymbol{x})} \left[\log p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{z})\right] \\ & - \mathbb{E}_{q(\boldsymbol{\theta}|\boldsymbol{d})} \left[\mathrm{KL}\left(q(\boldsymbol{z}|\boldsymbol{x}) || p(\boldsymbol{z}|\boldsymbol{\beta}, \boldsymbol{\theta})\right)\right] \,, \end{aligned} \quad (15)$$

where $\mathcal{L}_t$ is the same as used in (7). The main difference when compared with unconditional text generation lies in the usage of $p(\boldsymbol{y}|\boldsymbol{x}, \boldsymbol{z})$ and $q(\boldsymbol{z}|\boldsymbol{x})$, illustrated in Figure 1(b). The generation of $\boldsymbol{y}$ given $\boldsymbol{x}$ is not only dependent on a standard Seq2Seq model with attention (Nallapati et al., 2016), but also affected by $\boldsymbol{z}$ (*i.e.*, $\boldsymbol{z}_K$), which provides the high-level topic guidance.

3.3 Diversity Regularizer for NTM

Redundancy in inferred topics is a common issue existing in general topic models. In order to address this, it is straightforward to regularize the row-wise distance between paired topics to diversify the topics. Following Xie et al. (2015); Miao et al. (2017), we apply a topic diversity regularization while carrying out the inference.

Specifically, the distance between a pair of topics is measured by their cosine distance $a(\boldsymbol{\beta}_i, \boldsymbol{\beta}_j) = \arccos\left(\frac{|\boldsymbol{\beta}_i \cdot \boldsymbol{\beta}_j|}{|\,\boldsymbol{\beta}_i\,|_2 \, |\,\boldsymbol{\beta}_j\,|_2}\right)$. The mean angle of all pairs of T topics is $\phi = \frac{1}{T^2} \sum_i \sum_j a(\boldsymbol{\beta}_i, \boldsymbol{\beta}_j)$, and the variance is $\nu = \frac{1}{T^2} \sum_i \sum_j (a(\boldsymbol{\beta}_i, \boldsymbol{\beta}_j) - \phi)^2$. Finally, the topic-diversity regularization is defined as $R = \phi - \nu$.

4 Related Work

The VAE was proposed by Kingma and Welling (2013), and since then, it has been applied successfully in a variety of applications (Gregor et al., 2015; Kingma et al., 2014; Chen et al., 2017; Wang et al., 2018b; Shen et al., 2018). Focusing on text generation, the methods in Miao et al. (2017, 2016); Srivastava and Sutton (2017) represent texts as bag-of-words, and Bowman et al. (2015) proposed the usage of an RNN as the encoder and decoder, and found some negative results. In order to improve the performance, different convolutional designs (Semeniuta et al., 2017; Shen et al., 2017a; Yang et al., 2017) have been proposed. A VAE variant was further developed in Hu et al. (2017) to control the sentiment and tense of generated sentences. Additionally, the VAE has also been considered for conditional text generation tasks, including machine translation (Zhang et al., 2016), image captioning (Pu et al., 2016), dialogue generation (Serban et al., 2017; Shen et al., 2017b; Zhao et al., 2017) and text summarization (Li et al., 2017b; Miao and Blunsom, 2016). In particular, distinct from the above works, we propose the usage of a topic-dependent prior to explicitly incorporate topic guidance into the text-generation framework.

The idea of using learned topics to improve NLP tasks has been explored previously, including methods combining topic and neural language models (Ahn et al., 2016; Dieng et al., 2016; Lau et al., 2017; Mikolov and Zweig, 2012; Wang et al., 2017), as well as leveraging topic and word embeddings (Liu et al., 2015; Xu et al., 2018). Distinct from them, we propose the use of topics to guide the prior of a VAE, rather than only the language model (*i.e.*, the decoder in a VAE setup). This provides more flexibility in text modeling and also the ability to infer the posterior on latent codes, which could be useful for visualization and downstream tasks.

Neural abstractive summarization was pioneered in Rush et al. (2015), and it was followed and extended by Chopra et al. (2016). Currently the RNN-based encoder-decoder framework with attention (Nallapati et al., 2016; See et al., 2017) remains popular in this area. Attention models typ-

ically work as a keyword detector, which is similar to topic modeling in spirit. This fact motivated us to extend our topic-guided VAE model to text summarization.

5 Experiments

We evaluate our TGVAE on text generation and text summarization, and interpret its improvements both quantitatively and qualitatively.

5.1 Text Generation

Dataset We conduct experiments on three publicly available corpora: APNEWS, IMDB and BNC.[1] APNEWS[2] is a collection of Associated Press news articles from 2009 to 2016. IMDB is a set of movie reviews collected by Maas et al. (2011), and BNC (BNC Consortium, 2007) is the written portion of the British National Corpus, which contains excerpts from journals, books, letters, essays, memoranda, news and other types of text. For the three corpora, we tokenize the words and sentences, lowercase all word tokens, and filter out word tokens that occur less than 10 times. For the topic model, we remove stop words in the documents and exclude the top 0.1% most frequent words and also words that appear less than 100 documents. A summary statistics is provided in Table 1.

Evaluation We first compare the perplexity of our neural sequence model with a variety of baselines. Further, we evaluate BLEU scores on the generated sentences, noted as *test*-BLEU and *self*-BLEU. *test*-BLEU (higher is better) evaluates the quality of generated sentences using a group of real test-set sentences as the reference, and *self*-BLEU (lower is better) mainly measures the diversity of generated samples (Zhu et al., 2018).

Setup For the neural topic model (NTM), we consider a 2-layer feed-forward neural network to model $q(\boldsymbol{\theta}|\boldsymbol{d})$, with 256 hidden units in each layer; ReLU is used as the activation function. The hyper-parameter λ for the neural topic model diversity regularizer is fixed to 0.1 across all the experiments. All the sentences in the paragraph are used to obtain the bag-of-words presentation $\boldsymbol{d}$. The maximum number of words in a paragraph is set to 300. For the neural sequence model (NSM), we use bidirectional-GRU as the encoder and a standard GRU as the decoder. The hidden state of our GRU is fixed to 600 across all the three corpora. For the input sequence, we fix the sequence length to 30. In order to avoid overfitting, dropout with a rate of 0.4 is used in each GRU layer.

Baseline We test the proposed method with different numbers of topics (components in GMM) and different numbers of Householder flows ($i.e.$, K), and compare it with six baselines: (*i*) a standard language model (LM); (*ii*) a standard variational RNN auto-encoder (VAE); (*iii*) a Gaussian prior-based VAE with Householder Flow (VAE+HF); (*iv*) a standard LSTM language model with LDA as additional feature (LDA+LSTM); (*v*) Topic-RNN (Dieng et al., 2016), a joint learning framework which learns a topic model and a language model simultaneously; (*vi*) TDLM (Lau et al., 2017), a joint learning framework which learns a convolutional based topic model and a language model simultaneously.

Results The results in Table 3 show that the models trained with a VAE and its Householder extension does not outperform a well-optimized language model, and the KL term tends to be annealed with the increase of K. In comparison, our TGVAE achieves a lower perplexity upper bound, with a relative larger $\mathcal{U}_{KL}$. We attribute the improvements to our topic guided GMM model design, which provides additional topical clustering information in the latent space; the Householder flow also boosts the posterior inference for our TGVAE. We also observe consistent improvements with the number of topics, which demonstrates the efficiency of our TGVAE.

To verify the generative power of our TGVAE, we generate samples from our *topic-dependent* prior and compare various methods on the BLEU scores in Table 2. With the increase of topic numbers, our TGVAE yields consistently better *self*-BLEU and a boost over *test*-BLEU relative to standard VAE models. We also show a group of sampled sentences drawn from a portion of topics in Table 5. Our TGVAE is able to generate diverse sentences under topic guidance. When generating sentences under a mixture of topics, we draw multiple samples from the GMM and take $\boldsymbol{z}$ as the averaged sample.

Though this paper focuses on generating coherent topic-specific sentences rather than the learned topics themselves, we also evaluate the topic coherence (Lau et al., 2017) to show the rationality of our joint learning framework. We compute topic coher-

[1]These three datasets can be downloaded from https://github.com/jhlau/topically-driven-language-model.

[2]https://www.ap.org/en-gb/

Dataset	Vocabulary		Training			Development			Testing		
	LM	TM	# Docs	# Sents	# Tokens	# Docs	# Sents	# Tokens	# Docs	# Sents	# Tokens
APNEWS	$32,400$	$7,790$	$50K$	$0.7M$	$15M$	$2K$	$27.4K$	$0.6M$	$2K$	$26.3K$	$0.6M$
IMDB	$34,256$	$8,713$	$75K$	$0.9M$	$20M$	$12.5K$	$0.2M$	$0.3M$	$12.5K$	$0.2M$	$0.3M$
BNC	$41,370$	$9,741$	$15K$	$0.8M$	$18M$	$1K$	$44K$	$1M$	$1K$	$52K$	$1M$

Table 1: Summary statistics for APNEWS, IMDB and BNC.

Metric	Methods	APNEWS				IMDB				BNC			
		B-2	B-3	B-4	B-5	B-2	B-3	B-4	B-5	B-2	B-3	B-4	B-5
test-BLEU	VAE	0.564	0.278	0.192	0.122	0.597	0.315	0.219	0.147	0.479	0.266	0.169	0.117
	VAE+HF (K=1)	0.566	0.280	0.193	0.124	0.593	0.317	0.218	0.148	0.475	0.268	0.165	0.112
	VAE+HF (K=10)	0.570	0.279	0.195	0.123	0.610	0.322	0.221	0.147	0.483	0.270	0.169	0.110
	TGVAE (K=0, T=10)	0.582	0.320	0.203	0.125	0.627	0.362	0.223	0.159	0.517	0.282	0.181	0.115
	TGVAE (K=1, T=10)	0.581	0.326	0.202	0.124	0.623	0.358	0.224	0.160	0.519	0.282	0.182	0.118
	TGVAE (K=10, T=10)	0.584	0.327	0.202	0.126	0.621	0.357	0.223	0.159	0.518	0.283	0.173	0.119
	TGVAE (K=10, T=30)	0.627	0.335	0.207	0.131	**0.655**	0.369	**0.243**	**0.165**	0.528	**0.291**	0.182	0.119
	TGVAE (K=10, T=50)	**0.629**	**0.340**	**0.210**	**0.132**	0.652	**0.372**	0.239	0.160	**0.535**	0.290	**0.188**	**0.120**
self-BLEU	VAE	0.866	0.531	0.233	-	0.891	0.632	0.275	-	0.851	0.51	0.163	-
	VAE+HF (K=1)	0.865	0.533	0.241	-	0.899	0.641	0.278	-	0.854	0.515	0.163	-
	VAE+HF (K=10)	0.873	0.552	0.219	-	0.902	0.648	0.262	-	0.854	0.520	0.168	-
	TGVAE (K=0, T=10)	0.847	0.499	0.161	-	0.878	0.572	0.234	-	0.832	0.488	0.160	-
	TGVAE (K=1, T=10)	0.847	0.495	0.160	-	0.871	0.571	0.233	-	0.828	0.483	0.150	-
	TGVAE (K=10, T=10)	0.839	0.512	0.172	-	0.889	0.577	0.242	-	0.829	0.488	0.151	-
	TGVAE (K=10, T=30)	0.811	0.478	0.157	-	0.850	0.560	0.231	-	0.806	0.473	0.150	-
	TGVAE (K=10, T=50)	**0.808**	**0.476**	**0.150**	-	**0.842**	**0.559**	**0.227**	-	**0.793**	**0.469**	**0.150**	-

Table 2: *test*-BLEU (higher is better) and *self*-BLEU (lower is better) scores over three corpora.

Methods	APNEWS		IMDB		BNC	
	PPL	KL	PPL	KL	PPL	KL
LM	62.79	–	70.38	–	100.07	–
LDA+LSTM	57.05	–	69.58	–	96.42	–
Topic-RNN	56.77	–	68.74	–	94.66	–
TDLM	53.00	–	63.67	–	91.42	–
VAE	≤75.89	1.78	86.16	2.78	≤105.10	0.13
VAE+HF (K=1)	≤72.99	1.32	≤84.06	1.83	≤105.13	0.31
VAE+HF (K=10)	≤71.60	0.83	≤83.67	1.51	≤104.82	0.17
TGVAE (K=0, T=10)	≤56.12	2.73	≤62.99	3.99	≤92.32	3.40
TGVAE (K=1, T=10)	≤56.08	2.70	≤62.12	3.86	≤91.17	3.12
TGVAE (K=10, T=10)	≤55.77	2.69	≤62.22	3.94	≤91.19	2.99
TGVAE (K=10, T=30)	≤51.27	3.62	≤59.45	4.62	≤88.34	3.82
TGVAE (K=10, T=50)	≤**48.73**	3.55	≤**57.11**	5.02	≤**87.86**	4.57

Table 3: Perplexity and averaged KL scores over three corpora. KL in our TGVAE denotes $\mathcal{U}_{KL}$ in Eqn. (12).

ence using normalized PMI (NPMI). In practice, we average topic coherence over the top 5/10/15/20 topic words. To aggregate topic coherence score, we further average the coherence scores over topics. Results are summarized in Table 4.

5.2 Text Summarization

Dataset We further test our model for text summarization on two popular datasets. First, we follow the same setup as in Rush et al. (2015) and consider the GIGAWORDS corpus[3], which consists of 3.8M training pair samples, 190K validation samples and 1,951 test samples for evaluation. An input-summary pair consists of the first sentence and the headline of the source articles. We also evaluate various models on the DUC-2004 test set[4], which has 500 news articles. Different from GIGAWORDS, each article in DUC-2004 is paired with four expert-generated reference summaries. The length of each summary is limited to 75 bytes.

Methods	APNEWS T=50	IMDB T=50	BNC T=50
LDA (Blei et al., 2003)	0.125	0.084	0.106
TDLM (Lau et al., 2017)	0.149	0.104	0.102
Topic-RNN (Dieng et al., 2016)	0.134	0.103	0.102
TGVAE	**0.157**	**0.105**	**0.113**

Table 4: Topic coherence over APNEWS, IMDB and BNC.

Evaluation We evaluate the performance of our model with the ROUGE score (Lin, 2004), which counts the number of overlapping content between the generated summaries and the reference summaries, *e.g.*, overlapped n-grams. Following practice, we use F-measures of ROUGE-1 (RF-1), ROUGE-2 (RF-2) and ROUGE-L (RF-L) for GIGAWORDS and Recall measures of ROUGE-1 (RR-1), ROUGE-2 (RR-2) and ROUGE-L (RR-L) for DUC-2004.

Setup We have a similar data tokenization as we

[3]https://catalog.ldc.upenn.edu/ldc2012t21

[4]http://duc.nist.gov/duc2004

Data	Topic	Sentences
APNEWS	education	• the commission has approved a bill that would make state funding available for the city 's new school .
	animal	•the feline did n't survive fence hangars at the lake .
	crime	• the jury found the defense was not a <unk> , <unk> 's ruling and that the state 's highest court has been convicted of first-degree murder .
	weather	• forecasters say they 're still trying to see the national weather service watch for the latest forecast for friday evening .
	lottory	• she hopes the jackpot now exceeds $ 9 million .
	education+law	• an alabama law professor thomas said monday that the state's open court claims it takes an emotional matter about issuing child molesters based on religion.
	animal+medicine	• the study says the animal welfare department and others are not sure to make similar cases to the virus in the zoo.
IMDB	war	• after watching the movie , there is a great documentary about the war in the years of the israeli war .
	children	• the entire animation was great at times as to the readings of disney favorites .
	epsiode	• the show would have warranted for 25 episodes and it does help immediately .
	name	• she steals the other part where norma 's <unk> husband (crawford) (as at his part , sh*t for the road) .
	detective	• holmes shouted just to be as much as the movie 's last scene where there were <unk> pills to nab the <unk> .
	horror + negative	• the movie about a zombie is the worst movie i have ever seen.
	detective + children	• my favorite childhood is that rochester takes the character in jane's way, playing the one with hamlet.
BNC	medical	• here mistaking ' causes ' drugs as the problem although both economically ill patients arising from a local job will be in traumatic dangers .
	education	• he says the sale is given to five students ' award off : out at a laboratory after the three watts of the hours travelling in and chairman store the bank of the <unk> sutcliffe .
	religion	• schoolchildren will either go or back to church in his place every year in the savoy .
	entertainment	• 100 company and special lace with <unk> garland for tea our garden was filmed after a ceremony
	IT	• ibm also has shut all the big macs in the 60mhz ncube , represent on the acquisition and mips unix .
	environment + crime	• the earth's environmental protection agency said that the government was still being shut down by the police.
	education+entertainment	• the school is 55 and hosts one of a musician's theme charities festival.

Table 5: Generated sentences from given topics.

Sample of Summaries
D: a court here thursday sentenced a ##-year-old man to ## years in jail after he admitted pummelling his baby son to death to silence him while watching television . **R**: man who killed baby to hear television better gets ## years. **Seq2Seq**: man sentenced to ## years after the son 's death **Ours**: a court sentenced a man ## years in jail
D: european stock markets advanced strongly thursday on some bargain-hunting and gains by wall street and japanese shares ahead of an expected hike in us interest rates , dealers said **R**: european stocks bounce back UNK UNK with closing levels **Seq2Seq**: european stocks advance ahead of us interest rate hike **Ours**: european stocks rise on bargain-hunting, dealer said friday
D: the democratic people 's republic of korea whitewashed south korea in the women 's team semi-finals at the world table tennis championships here on sunday **R**: dpr korea sails into women 's team final **Seq2Seq**: dpr korea whitewash south korea in women 's team final **Ours**: dpr korea beat south korea in table tennis worlds

Table 6: Example generated summaries on GIGAWORDS. D is the source article, R means the reference summary, Seq2seq represents the summary generated from the Seq2Seq model.

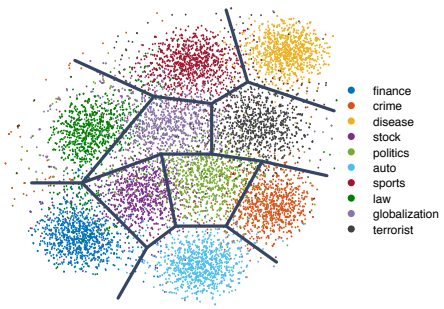


Figure 2: The t-SNE visualization of $1,000$ samples drawn from the learned topic-guided Gaussian mixture prior and they can be best viewed in color.

Methods	GIGAWORDS			DUC-2004		
	RF-1	**RF-2**	**RF-L**	**RR-1**	**RR-2**	**RR-L**
ABS	29.55	11.32	26.42	26.55	7.06	22.05
ABS+	29.78	11.89	26.97	28.18	8.49	23.81
RAS-LSTM	32.55	14.70	30.03	28.97	8.26	24.06
RAS-Elman	33.78	15.97	31.15	27.41	7.69	23.06
lvt2k-lsent	32.67	15.59	30.64	28.35	9.46	24.59
lvt5k-lsent	35.30	16.64	32.62	28.61	9.42	25.24
ASC+FSC	34.17	15.94	31.92	26.73	8.39	23.88
Seq2Seq	34.03	15.93	31.68	28.39	9.26	24.83
Var-Seq2Seq	34.00	15.97	31.85	28.11	9.24	24.86
Var-Seq2Seq-HF (K=1)	34.04	15.98	31.84	28.18	9.27	24.84
Var-Seq2Seq-HF (K=10)	34.22	16.10	32.13	28.78	9.11	24.96
TGVAE (K=0, T=10)	35.34	16.68	32.69	28.99	9.21	24.89
TGVAE (K=1, T=10)	35.35	16.70	32.64	29.02	9.24	24.93
TGVAE (K=10, T=10)	35.40	16.77	32.71	29.07	9.32	25.17
TGVAE (K=10, T=30)	35.59	17.18	32.89	29.38	**9.60**	25.22
TGVAE (K=10, T=50)	**35.63**	**17.27**	**33.02**	**29.65**	9.55	**25.38**

Table 7: Results on Gigawords and DUC-2004.

have in text generation. Additionally, for the vocabulary, we count the frequency of words in both the source article the target summary, and maintain the top 30,000 tokens as the source article and target summary vocabulary. For the NTM, we further remove top 0.3% words and infrequent words to get a topic model vocabulary in size of 8000. For the NTM, we follow the same setup as our text generation. In the NSM, we keep using bidirectional-GRU as the encoder and a standard GRU as the decoder. The hidden state is fixed to 400. An attention mechanism (Bahdanau et al., 2015) is applied in our sequence-to-sequence model.

Baseline We compare our method with the following alternatives: (*i*) a standard sequence-to-sequence model with attention (Bahdanau et al., 2015) (Seq2Seq); (*ii*) a model similar to our TGVAE, but without the usage of the topic-dependent prior and Householder flow (Var-Seq2Seq); and (*iii*) a model similar to our TGVAE, but without the usage of the topic dependent prior (Var-Seq2Seq-

Dataset	education	animal	crime	weather	lottory	terrorism	law	art	transportation	market
APNEWS	students	animals	murder	weather	mega	syria	lawsuit	album	airlines	zacks
	education	dogs	first-degree	corecasters	lottery	iran	appeals	music	rail	cents
	schools	zoo	shooting	winds	powerball	militants	justices	film	transit	earnings
	math	bear	sentenced	rain	gambling	afgan	constitutional	songs	bridge	revenue
	teachers	wildlife	gunshot	snow	jackpot	korea	judge	comedy	airport	income
IMDB	**war**	**children**	**epsiode**	**name**	**detective**	**ethic**	**action**	**horror**	**negative**	**japanese**
	aircraft	cinderella	season	crawford	holmes	porn	batman	horror	stupid	miike
	president	musical	episode	stanwyck	poirot	unfunny	king	zombie	horrible	kurosawa
	war	beatles	sandler	gable	christie	sex	chan	werewolf	sucks	sadako
	military	musicals	cartoons	powell	book	gay	li	candyman	waste	anime
	soldiers	disney	jokes	harlow	agatha	erotic	ninja	dracula	scary	takashi
BNC	**medical**	**education**	**religion**	**entertainment**	**IT**	**Law**	**facilities**	**crime**	**sports**	**environment**
	patients	award	church	film	unix	tax	bedrooms	police	cup	nuclear
	gastric	discipline	god	video	corp	coun	hotel	killed	league	emission
	cells	economic	art	album	software	lamont	restaurant	arrested	striker	dioxide
	oesophageal	research	theological	comedy	server	council	rooms	soldiers	season	pollution
	mucosa	institution	religious	movie	ibm	pensioners	situated	murder	goal	warming
GIGAWORDS	**terrorist**	**crime**	**finance**	**sports**	**law**	**stock**	**auto**	**disease**	**globalization**	**politics**
	palestinian	wounding	tael	scored	sentenced	seng	motor	flu	nuclear	republican
	arafat	killed	hk	rebounds	guilty	index	automaker	desease	eu	mccain
	yasser	roadside	gold	points	crimes	prices	toyota	virus	dpark	democrats
	abbas	injuring	cppcc	champion	court	taies	auto	bird	nato	barack
	israeli	crashed	cpc	beats	convicted	stock	ford	health	bilateral	presidential

Table 8: 10 topics learned from our model on APNEWS, IMDB, BNC and Gigawords.

HF).

Results The results in Table 7 show that our TGVAE achieves better performance than a variety of strong baseline methods on both GIGAWORDS and DUC-2004, demonstrating the practical value of our model. It is worthwhile to note that recently several much more complex CNN/RNN architectures have been proposed for abstract text summarization, such as SEASS (Zhou et al., 2017), ConvS2S (Gehring et al., 2017), and Reinforced-ConvS2S (Wang et al., 2018a). In this work, we focus on a relatively simple RNN architecture for fair comparison. In such a way, we are able to conclude that the improvements on the results are mainly from our topic-guided text generation strategy. As can be seen, though the Var-Seq2Seq model achieves comparable performance with the standard Seq2Seq model, the usage of Householder flow for more flexible posterior inference boosts the performance. Additionally, by combining the proposed topic-dependent prior and Householder flow, we yield further performance improvements, demonstrating the importance of topic guidance for text summarization.

To demonstrate the readability and diversity of the generated summaries, we present typical examples in Table 6. The words in blue are the topic words that appear in the source article but do not exist in the reference, while the words in red are neither in the reference nor in the source article. When the topic information is provided, our model is able to generate semantically-meaningful words which may not even exist in the reference summaries and the source articles. Additionally, with our topic-guided model, we can always generate a summary with meaningful initial words. These phenomena imply that our model supplies more insightful semantic information to improve the quality of generated summaries.

Finally, to demonstrate that our TGVAE learns interpretable topic-dependent GMM priors, we draw multiple samples from each mixture component and visualize them with t-SNE (Maaten and Hinton, 2008). As can be seen from Figure 2, we have learned a group of separable *topic-dependent* components. Each component is clustered and also maintains semantic meaning in the latent space, *e.g.*, the clusters corresponding to the topic "stock" and "finance" are close to each other, while the clusters for "finance" and "disease" are far away from each other. Additionally, to understand the topic model we have learned, we provide the top 5 words for 10 randomly chosen topics on each dataset (the boldface word is the topic name summarized by us), as shown in Table 8.

6 Conclusion

A novel text generator is developed, combining a VAE-based neural sequence model with a neural topic model. The model is an extension of conditional VAEs in the framework of unsupervised learning, in which the topics are extracted from the data with clustering structure rather than predefined labels. An effective inference method based on Householder flow is designed to encourage the complexity and the diversity of the learned topics. Experimental results are encouraging, across multiple NLP tasks.

References

Sungjin Ahn, Heeyoul Choi, Tanel Pärnamaa, and Yoshua Bengio. 2016. A neural knowledge language model. *arXiv preprint arXiv:1608.00318*.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *ICLR*.

Christian H Bischof and Xiaobai Sun. 1994. On orthogonal block elimination. *Preprint MCS-P450-0794, Mathematics and Computer Science Division, Argonne National Laboratory*.

David M Blei, Andrew Y Ng, and Michael I Jordan. 2003. Latent dirichlet allocation. *JMLR*.

BNC BNC Consortium. 2007. The British National Corpus, version 3 (BNC XML Edition). *Distributed by Bodleian Libraries, University of Oxford, on behalf of the BNC Consortium.*

Samuel R Bowman, Luke Vilnis, Oriol Vinyals, Andrew M Dai, Rafal Jozefowicz, and Samy Bengio. 2015. Generating sentences from a continuous space. *arXiv preprint arXiv:1511.06349*.

Changyou Chen, Chunyuan Li, Liqun Chen, Wenlin Wang, Yunchen Pu, and Lawrence Carin. 2017. Continuous-time flows for efficient inference and density estimation. *arXiv preprint arXiv:1709.01179*.

Liqun Chen, Shuyang Dai, Chenyang Tao, Haichao Zhang, Zhe Gan, Dinghan Shen, Yizhe Zhang, Guoyin Wang, Ruiyi Zhang, and Lawrence Carin. 2018. Adversarial text generation via feature-mover's distance. In *NeurIPS*.

Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*.

Sumit Chopra, Michael Auli, and Alexander M Rush. 2016. Abstractive sentence summarization with attentive recurrent neural networks. In *NAACL*.

Adji B Dieng, Chong Wang, Jianfeng Gao, and John Paisley. 2016. Topicrnn: A recurrent neural network with long-range semantic dependency. *arXiv preprint arXiv:1611.01702*.

Nat Dilokthanakul, Pedro AM Mediano, Marta Garnelo, Matthew CH Lee, Hugh Salimbeni, Kai Arulkumaran, and Murray Shanahan. 2016. Deep unsupervised clustering with Gaussian mixture variational autoencoders. *arXiv preprint arXiv:1611.02648*.

Jonas Gehring, Michael Auli, David Grangier, Denis Yarats, and Yann N Dauphin. 2017. Convolutional sequence to sequence learning. *arXiv preprint arXiv:1705.03122*.

Karol Gregor, Ivo Danihelka, Alex Graves, Danilo Jimenez Rezende, and Daan Wierstra. 2015. DRAW: A recurrent neural network for image generation. *arXiv preprint arXiv:1502.04623*.

Jiaxian Guo, Sidi Lu, Han Cai, Weinan Zhang, Yong Yu, and Jun Wang. 2017. Long text generation via adversarial training with leaked information. *arXiv preprint arXiv:1709.08624*.

Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P Xing. 2017. Toward controlled generation of text. In *ICML*.

Zhuxi Jiang, Yin Zheng, Huachun Tan, Bangsheng Tang, and Hanning Zhou. 2016. Variational deep embedding: An unsupervised and generative approach to clustering. *arXiv preprint arXiv:1611.05148*.

Yoon Kim, Sam Wiseman, Andrew C Miller, David Sontag, and Alexander M Rush. 2018. Semi-Amortized variational autoencoders. *arXiv preprint arXiv:1802.02550*.

Diederik P Kingma, Shakir Mohamed, Danilo Jimenez Rezende, and Max Welling. 2014. Semi-supervised learning with deep generative models. In *NIPS*.

Diederik P Kingma and Max Welling. 2013. Auto-encoding variational Bayes. *arXiv preprint arXiv:1312.6114*.

Jey Han Lau, Timothy Baldwin, and Trevor Cohn. 2017. Topically driven neural language model. *arXiv preprint arXiv:1704.08012*.

Jiwei Li, Will Monroe, Tianlin Shi, Sébastien Jean, Alan Ritter, and Dan Jurafsky. 2017a. Adversarial learning for neural dialogue generation. *arXiv preprint arXiv:1701.06547*.

Piji Li, Wai Lam, Lidong Bing, and Zihao Wang. 2017b. Deep recurrent generative decoder for abstractive text summarization. *arXiv preprint arXiv:1708.00625*.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. *Text Summarization Branches Out*.

Yang Liu, Zhiyuan Liu, Tat-Seng Chua, and Maosong Sun. 2015. Topical word embeddings. In *AAAI*.

Andrew L Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *ACL*.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-SNE. *JMLR*.

Yishu Miao and Phil Blunsom. 2016. Language as a latent variable: Discrete generative models for sentence compression. *arXiv preprint arXiv:1609.07317*.

Yishu Miao, Edward Grefenstette, and Phil Blunsom. 2017. Discovering discrete latent topics with neural variational inference. *arXiv preprint arXiv:1706.00359*.

Yishu Miao, Lei Yu, and Phil Blunsom. 2016. Neural variational inference for text processing. In *ICML*.

Tomas Mikolov and Geoffrey Zweig. 2012. Context dependent recurrent neural network language model. *SLT*.

Ramesh Nallapati, Bowen Zhou, Caglar Gulcehre, Bing Xiang, et al. 2016. Abstractive text summarization using sequence-to-sequence RNNs and beyond. *arXiv preprint arXiv:1602.06023*.

Yunchen Pu, Zhe Gan, Ricardo Henao, Xin Yuan, Chunyuan Li, Andrew Stevens, and Lawrence Carin. 2016. Variational autoencoder for deep learning of images, labels and captions. In *NIPS*.

Danilo Jimenez Rezende and Shakir Mohamed. 2015. Variational inference with normalizing flows. *arXiv preprint arXiv:1505.05770*.

Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. 2014. Stochastic backpropagation and approximate inference in deep generative models. *arXiv preprint arXiv:1401.4082*.

Alexander M Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. *arXiv preprint arXiv:1509.00685*.

Abigail See, Peter J Liu, and Christopher D Manning. 2017. Get to the point: Summarization with pointer-generator networks. *arXiv preprint arXiv:1704.04368*.

Stanislau Semeniuta, Aliaksei Severyn, and Erhardt Barth. 2017. A hybrid convolutional variational autoencoder for text generation. *arXiv preprint arXiv:1702.02390*.

Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron C Courville, and Yoshua Bengio. 2017. A hierarchical latent variable encoder-decoder model for generating dialogues. In *AAAI*.

Dinghan Shen, Qinliang Su, Paidamoyo Chapfuwa, Wenlin Wang, Guoyin Wang, Lawrence Carin, and Ricardo Henao. 2018. Nash: Toward end-to-end neural architecture for generative semantic hashing. *arXiv preprint arXiv:1805.05361*.

Dinghan Shen, Yizhe Zhang, Ricardo Henao, Qinliang Su, and Lawrence Carin. 2017a. Deconvolutional latent-variable model for text sequence matching. *arXiv preprint arXiv:1709.07109*.

Xiaoyu Shen, Hui Su, Yanran Li, Wenjie Li, Shuzi Niu, Yang Zhao, Akiko Aizawa, and Guoping Long. 2017b. A conditional variational framework for dialog generation. *arXiv preprint arXiv:1705.00316*.

Akash Srivastava and Charles Sutton. 2017. Autoencoding variational inference for topic models. *arXiv preprint arXiv:1703.01488*.

Xiaobai Sun and Christian Bischof. 1995. A basis-kernel representation of orthogonal matrices. *SIAM journal on matrix analysis and applications*.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *NIPS*.

Jakub M Tomczak and Max Welling. 2016. Improving variational auto-encoders using householder flow. *arXiv preprint arXiv:1611.09630*.

Li Wang, Junlin Yao, Yunzhe Tao, Li Zhong, Wei Liu, and Qiang Du. 2018a. A reinforced topic-aware convolutional sequence-to-sequence model for abstractive text summarization. *arXiv preprint arXiv:1805.03616*.

Wenlin Wang, Zhe Gan, Wenqi Wang, Dinghan Shen, Jiaji Huang, Wei Ping, Sanjeev Satheesh, and Lawrence Carin. 2017. Topic compositional neural language model. *arXiv preprint arXiv:1712.09783*.

Wenlin Wang, Yunchen Pu, Vinay Kumar Verma, Kai Fan, Yizhe Zhang, Changyou Chen, Piyush Rai, and Lawrence Carin. 2018b. Zero-shot learning via class-conditioned deep generative models. In *AAAI*.

Pengtao Xie, Yuntian Deng, and Eric Xing. 2015. Diversifying restricted Boltzmann machine for document modeling. In *KDD*.

Hongteng Xu, Wenlin Wang, Wei Liu, and Lawrence Carin. 2018. Distilled wasserstein learning for word embedding and topic modeling. In *NIPS*.

Zichao Yang, Zhiting Hu, Ruslan Salakhutdinov, and Taylor Berg-Kirkpatrick. 2017. Improved variational autoencoders for text modeling using dilated convolutions. *arXiv preprint arXiv:1702.08139*.

Lantao Yu, Weinan Zhang, Jun Wang, and Yong Yu. 2017. SeqGAN: Sequence generative adversarial nets with policy gradient. In *AAAI*.

Biao Zhang, Deyi Xiong, Jinsong Su, Hong Duan, and Min Zhang. 2016. Variational neural machine translation. *arXiv preprint arXiv:1605.07869*.

Ruiyi Zhang, Changyou Chen, Zhe Gan, Wenlin Wang, Liqun Chen, Dinghan Shen, Guoyin Wang, and Lawrence Carin. 2018. Sequence generation with guider network. *arXiv preprint arXiv:1811.00696*.

Ruiyi Zhang, Chunyuan Li, Changyou Chen, and Lawrence Carin. 2017a. Learning structural weight uncertainty for sequential decision-making. *arXiv preprint arXiv:1801.00085*.

Yizhe Zhang, Zhe Gan, Kai Fan, Zhi Chen, Ricardo Henao, Dinghan Shen, and Lawrence Carin. 2017b. Adversarial feature matching for text generation. *arXiv preprint arXiv:1706.03850*.

Tiancheng Zhao, Ran Zhao, and Maxine Eskenazi. 2017. Learning discourse-level diversity for neural dialog models using conditional variational autoencoders. *arXiv preprint arXiv:1703.10960*.

Qingyu Zhou, Nan Yang, Furu Wei, and Ming Zhou. 2017. Selective encoding for abstractive sentence summarization. *arXiv preprint arXiv:1704.07073*.

Yaoming Zhu, Sidi Lu, Lei Zheng, Jiaxian Guo, Weinan Zhang, Jun Wang, and Yong Yu. 2018. Texygen: A benchmarking platform for text generation models. *arXiv preprint arXiv:1802.01886*.