Dispersed Exponential Family Mixture VAEs for Interpretable Text Generation

Wenxian Shi [1]   Hao Zhou [1]   Ning Miao [1]   Lei Li [1]

Abstract

Interpretability is important in text generation for guiding the generation with interpretable attributes. Variational auto-encoder (VAE) with Gaussian distribution as prior has been successfully applied in text generation, but it is hard to interpret the meaning of the latent variable. To enhance the controllability and interpretability, one can replace the Gaussian prior with a mixture of Gaussian distributions (GM-VAE), whose mixture components could be related to some latent attributes of data. Unfortunately, straightforward variational training of GM-VAE leads the *mode-collapse* problem. In this paper, we find that mode-collapse is a general problem for VAEs with exponential family mixture priors. We propose DEM-VAE, which introduces an extra *dispersion* term to induce a well-structured latent space. Experimental results show that our approach does obtain a well structured latent space, with which our method outperforms strong baselines in interpretable text generation benchmarks.

1. Introduction

Text generation is one of the most challenging problems in Natural Language Processing (NLP), which is widely applied to tasks such as machine translation (Brown et al., 1993) and dialog system (Young et al., 2013). Though generation quality is the main concern of most work, the interpretability for generation process is also of great importance. Interpretable generation models could explore the latent data structures, such as topics (Wang et al., 2019) and dialog actions (Zhao et al., 2018b), and uses them to guide text classification and further generation. Among deep generative models, variational auto-encoder (Kingma & Welling, 2013; Rezende et al., 2014, VAE) is especially

[1]ByteDance AI lab. Correspondence to: Wenxian Shi <shiwenxian@bytedance.com>, Hao Zhou <zhouhao.nlp@bytedance.com>, Ning Miao <miaoning@bytedance.com>, Lei Li <lileilab@bytedance.com>.

suitable for interpretable text generation because it maps sentences to a regularized latent variable z, which can be used to derive interpretable structures (Bowman et al., 2016; Miao et al., 2016; Semeniuta et al., 2017; Xu & Durrett, 2018).

However, the continuous latent variable z of vanilla VAE makes it difficult to interpret discrete attributes, such as topics and dialog actions. Recently, Zhao et al. (2018b) propose to replace the continuous latent variable with a discrete one for better interpretation in generating dialog, where the discrete latent variable represents the dialog actions in their system, producing promising results even in an unsupervised setting. Unfortunately, the expressiveness of VAE with only a discrete latent variable c is very limited. It only contains $\log(\#c)$ bits of information, where $\#c$ is the number of all possible values of c. So it is impossible for discrete VAE to express the complicated sentence space.

To solve the above concern, Gaussian mixture VAE (GM-VAE) offers a natural choice, which enjoys the benefits of both discrete and continuous latent space. Each component c represents a discrete attribute, while continuous latent variable z in each component represents different sentences with the same attribute. However, GM-VAE is prone to *mode-collapse*, which makes it difficult to train. In other words, different components tend to have very close means and variances after training, which makes GM-VAE degenerate to vanilla VAE with only one Gaussian component (Fig. 1). As a result, GM-VAE fails to capture the multi-modal data structure, and cannot effectively utilize the discrete latent variables. For example, as illustrated in Fig. 1a, utterances ask about the weather and requesting appointments are mapped into the same mode. In this paper, we theoretically demonstrate that mode-collapse does not only occur in GM-VAE, which is a general problem for VAEs with exponential family mixture priors (EM-VAE). We find that the problem is intrinsically caused by a "dispersion" term in the evidence lower bound (ELBO), which makes it extremely easy to fall into the solution of mode collapse.

To address the mode-collapse problem, we propose Dispersed EM-VAE (DEM-VAE), which introduces an extra *dispersion* term in the training objective (Fig. 1b) for overcoming the contraction tendency introduced by ELBO. Ex-
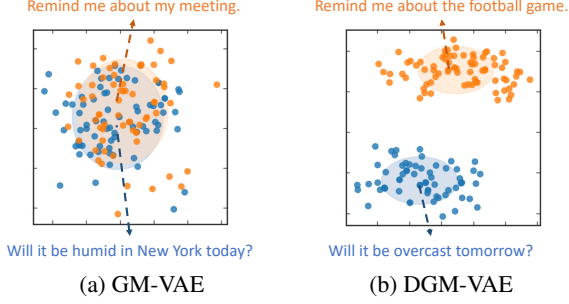
Figure 1. Latent space learned by GM-VAE and our proposed DEM-VAE with a specific Gaussian mixture prior (named DGM-VAE) for dialog intention recognition. Notice that our proposed DGM-VAE avoids mode-collapse of GM-VAE. As a result, requests with different intentions are clearly separated into two clusters.

Figure 2. Graphical models for various VAEs. z and c are continuous and discrete latent variables, respectively. x is observed data. The solid lines are conditional dependencies and dashed lines represent variational posteriors.

perimental results show that our proposed DEM-VAE alleviates the mode-collapse problem effectively. Furthermore, DEM-VAE outperforms strong baselines in interpretable text generation benchmarks. Although DEM-VAE moderately decreases the sentence likelihood because of the inclusion of external training term, it outputs sentences with significantly better quality in aspects of other evaluation metrics such as rPPL and BLEU scores.

2. Related Work

VAEs for Language Generation. Variational auto-encoders are proposed by Kingma & Welling (2013, VAEs) and (Rezende et al., 2014), and applied by Bowman et al. (2016) for natural language generation. VAEs are extended by many following works in various specific language generation tasks, such as dialog generation (Serban et al., 2017; Wen et al., 2017; Zhao et al., 2017b; 2018b), summarization (Li et al., 2017a) and other natural language generation tasks (Miao et al., 2016; Zhang et al., 2016; Semeniuta et al., 2017; Gupta et al., 2018; Xu & Durrett, 2018).

Additionally, Wen et al. (2017) and Zhao et al. (2018b) propose to replace the continuous latent variable with a discrete one for interpretable sentence generation. Kingma et al. (2014) propose the *semi*-VAE for semi-supervised learning. This model is then adopted by Hu et al. (2017); Zhou & Neubig (2017) for style-transfer and labeled sequence transduction, respectively. Different from GM-VAE, continuous and discrete latent variables in *semi*-VAE are independent.

Gaussian Mixture VAEs. Using Gaussian mixture models as priors in VAEs is not new. Gaussian mixture variational auto-encoder has been used in the unsupervised clustering (Dilokthanakul et al., 2016; Jiang et al., 2017), obtaining promising results. Wang et al. (2019) used GMM as priors for topic-guided text generation. In this work, we apply
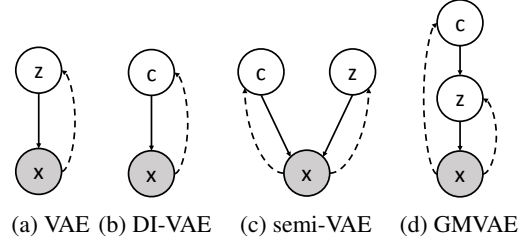
GM-VAE for interpretable text generation and propose the DGM-VAE to address the mode-collapse problem according to our theoretical analysis.

KL Collapse vs. Mode Collapse. The vanilla VAE models usually suffer from the KL collapse problem (Bowman et al., 2016) in language generation, in which the KL regularization term will quickly collapse to 0. A line of following work (Bowman et al., 2016; Zhao et al., 2017b; 2018b; Higgins et al., 2017; He et al., 2018; Li et al., 2019) is proposed to avoid the KL collapse problem.

These methods are different from ours in two-folds. First, the problem to be solved is different. Our method aims to address the mode-collapse in the mixture of exponential family VAEs, while these methods try to fix the posterior collapse in general VAEs. Mode collapse and posterior collapse are not the same. Posterior collapse means that the posterior distributions of all latent variables collapse to their priors, while mode-collapse means the multiple modes of prior collapses to one mode. Mode-collapse might also occur when the posterior does not collapse. Second, although our method and previous methods of solving KL collapse have similar solutions that find the problem-causing term in the objective and weakens it, a heuristic dispersion term instead of the whole KL term is introduced according to our theoretical analysis. The dispersion term is explicitly related to the difference between prior components and is specifically effective for the mode collapse problem.

3. Approach

We first describe the VAEs with mixture of exponential family priors for text generation, and investigate the mode-collapse issue in them. Based on the results of the investigation, we propose dispersed exponential family mixture VAEs to fix this issue, among which dispersed Gaussian mixture VAE (DGM-VAE) is particularly exemplified.

3.1. Mixture of Exponential Family VAEs

Mixture of Exponential Family VAEs are variational auto-encoders that adopts the mixture of exponential family distributions as its prior. GM-VAE is the most popular exponential family mixture VAE, whose prior is a mixture of Gaussian (Bishop, 2006). GM-VAE employs a discrete latent variable c to represent the mixture components, and a continuous latent variable z dependent on c. In this model, the marginal likelihood of a sentence x is:

$$p(x) = \int \sum_c p_\eta(z, c) p_\theta(x|z) dz, \qquad (1)$$

in which θ is the parameters of generation model which generates x from z. $p_\eta(z, c)$ is the mixture *prior* distribution with parameters η and can be factorized by $p(z, c) = p(c)p_\eta(z|c)$. Intuitively, $p(c)$ could be assumed as a uniform distribution; while $p_\eta(z|c)$ is an exponential family distribution, such as Gaussian, of the corresponding c-th component.

Fig. 2 compares the graphical models of VAE variants. The vanilla VAE (Kingma & Welling, 2013) uses a latent continuous variable, while the discrete VAE models (Zhao et al., 2018b) use a discrete one. The semi-VAE (Kingma et al., 2014) combines independent discrete and continuous latent variables, and the GM-VAE (Dilokthanakul et al., 2016; Jiang et al., 2017) impose dependency between them.

Testing. During testing, a mixture component c is first chosen according to the prior distribution $p(c)$. Then the variable z is sampled from the chosen component $p_\eta(z|c)$. As in Bowman et al. (2016), a *generation network* takes z as input and generate the sentence x through a decoder $p_\theta(x|z)$.

Training. Optimizing and inference for Eq. 1 is difficult. Following previous work of Kingma & Welling (2013) and Rezende et al. (2014), we use a variational *posterior* distribution $q_\phi(z, c|x)$ with parameters ϕ to approximate the real posterior distribution $p(z, c|x)$. With the mean field approximation (Xing et al., 2003), $q_\phi(z, c|x)$ can be factorized as: $q_\phi(z, c|x) = q_\phi(z|x)q_\phi(c|x)$.

The posterior $q_\phi(z|x)$ could be assumed as a multivariate Gaussian distribution, whose mean $\mu_\phi(x)$ and variance $\sigma_\phi^2(x)$ are obtained through a neural network (*recognition network*). $q_\phi(c|x)$ could be implemented by a neural network classifier.

Instead of optimizing the marginal likelihood in Eq. 1, we maximize a evidence lower bound (ELBO). The ELBO can be decomposed as the summation of a reconstruction term and regularization terms for c and z, respectively:

$$\begin{aligned}
\text{ELBO} &= \mathbb{E}_{q_\phi(z,c|x)}\left(\log p_{\theta,\eta}(x, c, z) - \log q_\phi(z, c|x)\right) \\
&= \underbrace{\mathbb{E}_{q_\phi(z|x)q_\phi(c|x)} \log p_\theta(x|z)}_{\text{Reconstruction gain } (\mathcal{R}_{\text{rec}})} \\
&+ \underbrace{\mathbb{E}_{q_\phi(z|x)q_\phi(c|x)}\left(\log p(c) - \log q_\phi(c|x)\right)}_{\text{Regularization on c } (\mathcal{R}_c)} \\
&+ \underbrace{\mathbb{E}_{q_\phi(z|x)q_\phi(c|x)}\left(\log p_\eta(z|c) - \log q_\phi(z|x)\right)}_{\text{Regularization on z } (\mathcal{R}_z)}.
\end{aligned}$$

All parameters including θ, ϕ and η could be jointly trained with reparameterization tricks (Kingma & Welling, 2013; Jang et al., 2016)

However, exponential family mixture VAEs often encounter *mode-collapse*, where all components of mixture prior degenerate to one distribution.

3.2. Mode-Collapse Problem

We further investigate the ELBO objective function to analyze mode-collapse. To this end, we present that the regularization terms of exponential family mixture VAE's ELBO, $\mathcal{R}_c$ and $\mathcal{R}_z$, are responsible for the mode collapse problem. We only give explanations and remarks of the proof, with the details included in the supplementary materials.

According to the definition of exponential family, we represent the probability density function of c-th component in mixture prior by the natural parameters

$$p_\eta(z|c) = \exp(<\boldsymbol{\eta}_c, \boldsymbol{\phi}(z)> - A(\boldsymbol{\eta}_c)), \qquad (2)$$

in which $\boldsymbol{\phi}(z)$ is a vector of functions named sufficient statistics, and $\boldsymbol{\eta}_c$ is the corresponding parameters vector. $<\boldsymbol{\eta}_c, \boldsymbol{\phi}(x)>$ means the inner-product of vector $\boldsymbol{\eta}_c$ and $\boldsymbol{\phi}$. $A(\boldsymbol{\eta}_c)$ is the log-partition function for normalizing the probability density. Taking Gaussian distribution as an example, $\boldsymbol{\phi}(z) = [z, z^2]$, $\boldsymbol{\eta}_c = [\eta_1, \eta_2] = [\frac{\mu_c}{\sigma_c^2}, -\frac{1}{2\sigma_c^2}]$ and $A(\eta_1, \eta_2) = -\frac{\eta_1^2}{4\eta_2} - \frac{1}{2}\log(-2\eta_2)$.

Plugging the definition of Eq. 2 in regularization $\mathcal{R}_z$, $\mathcal{R}_z$ could be re-written as two terms: an "average" KL regularization term and a "dispersion" term:

$$\begin{aligned}
\mathbb{E}_{q_\phi(z|x)q_\phi(c|x)} \log \frac{p_\eta(z|c)}{q_\phi(z|x)} &= \underbrace{-\text{KL}(q_\phi(z|x)||\hat{p}_{\overline{\eta}}(z|x)))}_{\text{Average } \mathcal{R}_z} \\
&- \underbrace{\left(\mathbb{E}_{q_\phi(c|x)} A(\boldsymbol{\eta}_c) - A(\mathbb{E}_{q_\phi(c|x)} \boldsymbol{\eta}_c)\right)}_{\text{Dispersion term } \mathcal{L}_d}.
\end{aligned}$$

The "average" KL regularization term is a KL divergence between posterior $q_\phi(z|x)$ and an "average" prior: $\hat{p}_{\overline{\eta}}(z|x) = \exp\left(\langle \overline{\boldsymbol{\eta}}, \boldsymbol{\phi}(x) \rangle - A(\overline{\boldsymbol{\eta}})\right)$, which is an exponential family

distribution parameterized by "average" parameters $\overline{\eta} = \mathbb{E}_{q_\phi(c|x)} \, \eta_c$. Note that for exponential families, the domain of the parameters η is an convex set, thus the $\overline{\eta}$ is feasible parameters.

The "dispersion" term is respect to priors parameters and $q_\phi(c|x)$. In the following, we demonstrate two properties of the dispersion term inducing the mode-collapse.

Property 1. *Dispersion term $\mathcal{L}_d = \mathbb{E}_{q_\phi(c|x)} A(\eta_c) - A(\mathbb{E}_{q_\phi(c|x)} \, \eta_c) \geq 0$ and it gets to zero when parameters η_c of different components c collapse to the same one, when $q_\phi(c|x)$ does not have one-hot probability mass.*

For minimal representation of exponential family[1], the Property 1 could be easily proven by the strictly convexity of log-partition function (Wainwright et al., 2008). Besides, the $\mathcal{R}_c$ prevent the $q_\phi(c|x)$ to be one-hot because the $p(c)$ is always assumed to be uniform.

Property 2. *Minimizing the dispersion term $\mathcal{L}_d$ makes the weighted variance of prior parameters, defined as*

$$Var_{q(c|x)}\eta_c = Tr\big(\mathbb{E}_{q(c|x)}[(\eta_c - \mathbb{E}_{q(c|x)} \, \eta_c)^T (\eta_c - \mathbb{E}_{q(c|x)} \, \eta_c)]\big),$$

smaller.

Because of the convexity of partition function, the gradients of $\mathcal{L}_d$ and $Var_{q(c|x)}\eta_c$ are along the same direction,

$$(\nabla_{\eta_c} \mathcal{L}_d)^T \cdot (\nabla_{\eta_c} Var_{q(c|x)}\eta_c) \sim$$
$$\big[\nabla A|_{\eta_c} - \nabla A|_{E_{q(c|x)}\eta_c}\big]^T \cdot \big[\eta_c - (E_{q(c|x)}\eta_c)\big] \geq 0,$$

which means minimizing the dispersion term will implicitly minimize the variance of prior parameters and makes these parameters tend to be the same.

Thus, according to the aforementioned two properties, the presence of dispersion term in the ELBO objective encourages the parameters of all mixture components to be as indistinguishable as possible and induces the mode-collapse.

In the following, we take Gaussian mixture and categorical mixture models as examples to show their specific dispersion term, and illustrate how their dispersion value are related to the degree of mode-collapse.

$\mathcal{L}_d$ **in Gaussian Mixture VAE.** We take the uni-variate Gaussian mixture VAE with known variance σ as an example. The natural parameter of Gaussian distribution is $\eta = \mu/\sigma$, in which μ is known as the mean, and the log-partition function $A(\eta) = \eta^2/2$. As a result, the dispersion term could be written as $\frac{1}{2\sigma^2}[\mathbb{E}_{q_\phi(c|x)} \mu_c^2 - (\mathbb{E}_{q_\phi(c|x)} \mu_c)^2]$, which is proportional to the variance of μ_c.

$\mathcal{L}_d$ **in Categorical Mixture VAE.** Considering the simplest categorical mixture VAE (CM-VAE), Bernoulli mixture

[1]Sufficient statistics are linearly independent to each other.

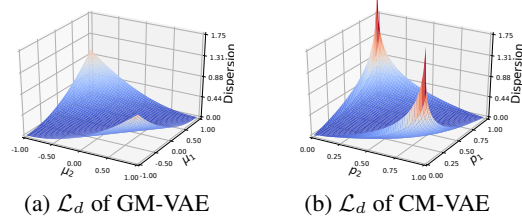(a) $\mathcal{L}_d$ of GM-VAE   (b) $\mathcal{L}_d$ of CM-VAE

Figure 3. Dispersion term $\mathcal{L}_d$ of Gaussian and categorical (Bernoulli) mixture VAEs with respect to the parameters of two prior components.

VAE, the only natural parameter $\eta_c = \log \frac{p_c}{1-p_c}$ and the $A(\eta_c) = \log(1 + e^{\eta_c})$. As a result, the dispersion term is $-\log(\prod_c p_c^{q_\phi(c|x)} + \prod_c (1-p_c)^{q_\phi(c|x)})$.

Visualization of these two dispersion term with respect to their prior parameters are shown in Fig. 3. In Fig. 3a, μ_1 and μ_2 are parameters of two components for one-dimensional GM-VAE, while in Fig. 3b p_1 and p_1 are parameters of two components for CMVAE. $q_\phi(c|x)$ is assumed to be uniform. It is illustrated that the more similar the parameters are, the lower the dispersion item will be.

3.3. Dispersed Exponential Family Mixture VAEs

In this section, we propose the Dispersed Exponential Family Mixture VAEs (DEM-VAE), which is a simple yet effective way to avoid the mode-collapse problem. We takes the Gaussian mixture VAE (GM-VAE) as a specific example, and its corresponding proposed model is named as DGM-VAE.

According to the theoretical insights in Sec. 3.2, we include an extra positive *dispersion term* to balance the mode collapse from ELBO. The objective of DEM-VAE $\mathcal{L}(\theta; x)$ for x sampled from the dataset D is:

$$\mathcal{L}(\theta; x) = \text{ELBO} + \beta \cdot \mathcal{L}_d,$$
$$\mathcal{L}_d = \mathbb{E}_{q_\phi(c|x)} A(\eta_c) - A(\mathbb{E}_{q_\phi(c|x)} \, \eta_c).$$

For mixture of Gaussian, we have $\eta_c = [\eta_1, \eta_2] = [\frac{\mu_c}{\sigma_c^2}, -\frac{1}{2\sigma_c^2}], A(\eta_c) = -\frac{\eta_1^2}{4\eta_2} - \frac{1}{2}\log(-2\eta_2)$. $\mathcal{L}_d$ with a hyper-parameter β is proposed to regularize the dispersion trends of mixture components. We can tune β to make a trade-off between variance and concentration degree of prior components.

The final objective of DEM-VAE could be:

$$\frac{1}{|D|} \sum_{x \sim D} \Big(\mathbb{E}_{q_\phi(z|x)} \log p_\theta(x|z) - \text{KL}(q_\phi(c|x)||p(c))$$
$$- \text{KL}(q_\phi(z|x)||\hat{p}_{\overline{\eta}}(z|x))) - (1-\beta)\mathcal{L}_d \Big).$$

Additional Mutual Information Term. As introduced

by previous works (Chen et al., 2016; Zhao et al., 2017a; 2018b), adding mutual information term to ELBO could enhance the interpretability and alleviate the KL-collapse. Our method could be further improved in interpretability through adding the mutual information term

$$\mathcal{L}_{\mathrm{mi}} = \mathcal{H}(c) - \mathcal{H}(c|x) = \mathbb{E}_x \, \mathbb{E}_{q_\phi(c|x)} (\log q_\phi(c|x) - \log q_\phi(c))$$

to the objective. $q_\phi(c)$ could be estimated by $\mathbb{E}_x \, q_\phi(c|x)$ within mini-batch (Zhao et al., 2018b).

In the following, we take the Guassian mixture as an example and demonstrate the architecture of DGM-VAE for text generation. Except for the learning objective, DGM-VAE has similar architecture as GM-VAE. It consists of a encoder for learning posterior and a decoder for generation.

Encoder. Recurrent neural networks such as GRU (Chung et al., 2014) as recognition networks encode sentences into compact hidden states. The parameters of posterior $q_\phi(z|x)$ and $q_\phi(c|x)$ are obtained based on hidden states. For example, mean μ_ϕ and variance σ_ϕ^2 of the posterior distribution $q_\phi(z|x)$ (assumed as a multivariate diagonal Gaussian) are obtained from two affine transformations. $q_\phi(c|x)$ could be modelled as a non-linear classifier taking the last hidden states as input.

Decoder. In the decoding phase, we first sample a z from a mixture priors (in testing) or from posterior (in training) by the reparameterization trick (Kingma & Welling, 2013). The sentences will be generated on a recurrent neural language model fashion (generation networks), with the z as the initialized hidden state. z could also be concatenated with embedding as input at each decoding step.

Interpretable Dialog Generation. We follow the same approach of DI-VAE (Zhao et al., 2018b) for interpretable dialog generation. The approach could be extended to other scenarios of interpretable generations, but we only validate our DGM-VAE on dialog for comparing with Zhao et al. (2018b).

Specifically, in dialog generation, we generate response r given the dialog context y. A DGM-VAE model is pre-trained in all utterances of the training set to capture the interpretable facts (component latent variable c) such as dialog actions or intentions. In training, a hierarchical recurrent encoder-decoder model (HRED) with attention (Sordoni et al., 2015; Serban et al., 2016) $p_\theta(r|z, y)$ is trained to generate the response. Here z is obtained from the pre-trained recognition network $q_\phi(z|r)$ of DGM-VAE and then fed into the decoder. A policy network $p_\pi(c|y)$ is trained jointly to predict c sampled from $q_\phi(c|r)$ in order to predict c in the testing stage.

4. Experiments

In this section, we empirically test the generation quality and interpretable ability of two special cases of DEM-VAE, Dispersed GM-VAE (DGM-VAE) and Dispersed Multivariate Categorical Mixture VAE (DCM-VAE), on standard benchmarks. Results on dialog generation, attribute detection, and text classification demonstrate the superiority of DGM-VAE.

4.1. Setup

We conduct experiments following Zhao et al. (2018b). For generation quality, we use the Penn Treebanks (Marcus et al., 1993, PTB) pre-processed by Mikolov (Mikolov et al., 2010) as the benchmark. For interpretability, we use the Daily Dialogs (Li et al., 2017b, DD) and the Stanford Multi-Domain Dialog (Eric et al., 2017, SMD) datasets. DD is a chat-oriented dataset containing 13,118 multi-turn dialogs, annotated with dialog actions and emotions. SMD contains 3,031 human-Woz, task-oriented dialogs collected from 3 different domains (navigation, weather and scheduling). In addition to unsupervised text generation tasks, we make use of emotion and action labels in DD as supervision and conduct supervised text generation experiments.

We compare our model with the following baselines: 1) RNNLM, language model (Mikolov et al., 2010) implemented by GRU (Cho et al., 2014); 2) AE, auto-encoders (Vincent et al., 2010) without latent space regularization; 3) DAE, auto-encoders with discrete latent space; 4) VAE, the vanilla VAE (Kingma & Welling, 2013) with only continuous latent variable and normal distribution prior; 5) DVAE, VAE with discrete latent variables; 6) DI-VAE, a DVAE variant (Zhao et al., 2018b) adding extra mutual information term; 7) *semi*-VAE, semi-supervised VAE model proposed by Kingma et al. (2014) with independent discrete and continuous latent variables; 8) GM-VAE, vanilla GM-VAE models as introduced in 3.1. Gumbel-softmax (Jang et al., 2016) is used for reparameterization in VAE variants with discrete latent variables.

The encoder and decoder in all models are implemented with single-layer GRU (Chung et al., 2014), with the hidden size as 512. For VAEs with discrete latent variables, multiple independent variables are adopted in order to increase model capacity. For unsupervised text generation, the dimension of discrete latent variables is set to 5 while the number of discrete latent variables is set to 20, 3 and 3 for PTB, DD, and SMD. The total dimension of continuous latent space is set to 40 for PTB, 15 for DD and 48 for SMD. For supervised text generation, the discrete variable number is 30, the dimension of each variable is set to 8 and the number of mixture components is set to 30. KL annealing with logistic weight function $1/(1 + \exp(-0.0025(\text{step} - 2500)))$ is adopted for all VAE variants. For GM-VAE, the KL anneal-
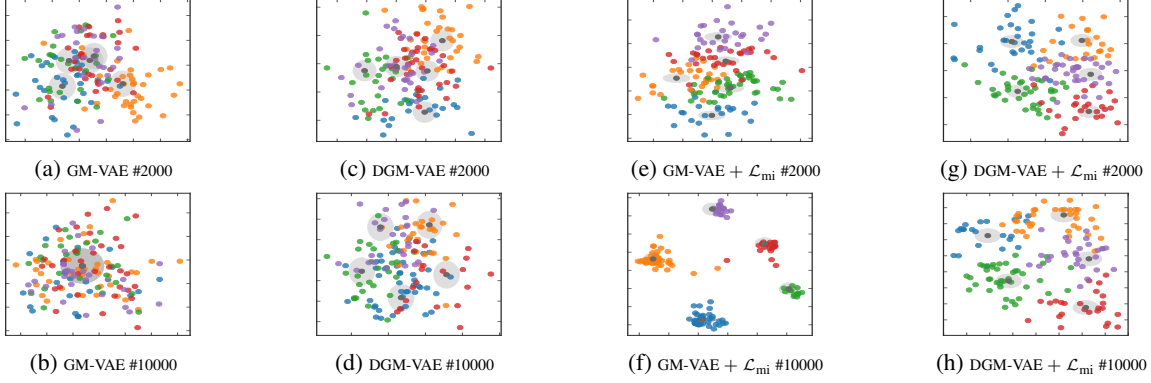
Figure 4. Visualization of the mode collapse problem in DD dataset for GM-VAE, GM-VAE adding $\mathcal{L}_{\mathrm{mi}}$ term, DGM-VAE and DGM-VAE adding $\mathcal{L}_{\mathrm{mi}}$. Gaussian mixture priors are represented by grey points (mean) and circles (variance). The mean of posteriors are marked as colored points (colors are associated with discrete latent variables).

ing is applied in the whole KL term. All hyper-parameters including β are chosen according to the reverse perplexity (language generation task) or BLEU scores (dialog generation task) in the validation set. Details of hyper-parameters are included in the supplementary.

4.2. Effects of DGM-VAE on Mode-Collapse

We illustrate the effectiveness of DGM-VAE to alleviate the mode-collapse problem. We train GM-VAE and DGM-VAE in utterances on the DD dataset, and randomly sample 300 points from test data at 2,000 and 10,000 training steps, respectively. The dimension of latent space is set to 2 for visualization. As in Fig. 4, the mean and variance of each Gaussian component are indicated by grey points and circles, respectively. The means of posteriors are marked as colored points (points with different discrete latent variables are associated with different colors).

It can be seen that, after 10,000 training steps, the vanilla GM-VAE degenerates into uni-Gaussian VAE, with the same mean values of all Gaussian components (Fig. 4a and 4b). DGM-VAE gives promising results as shown in Fig. 4d and 4h, in which different components of the GMM are dispersed and cluster data points into multiple modes well. As shown in Fig. 4f, adding mutual information to GM-VAE indeed helps to alleviate the mode-collapse. However, the posterior points are quite concentrated to the priors, which makes the latent space degenerates into a discrete one so that the model cannot enjoy the high capacity and diversity of continuous variables.

4.3. Language Generation Performance

We evaluate the performance of language generation on PTB in Tab. 1, comparing DGM-VAE ($\beta = 0.8$) with baselines described in Sec. 4.1. The test set of PTB is also included for comparison of text fluency.

We include four metrics to evaluate the generation performances: reverse perplexity (**rPPL**), **BLEU** (Papineni et al., 2002), word-level KL divergence (**wKL**) and negative log-likelihood (**NLL**). Reverse perplexity is the perplexity of an LSTM language model (Merity et al., 2017) trained on the synthetic data sampled from priors of VAE variants, and evaluated on the test set (Zhao et al., 2018a). Lower rPPL indicates that the sampled sentences are more diverse and fluent. The BLEU score between input and reconstructed sentences reflects the ability of reconstruction. Word-level KL divergence between word frequencies calculated in generated data and training data shows the quality of generation. Negative log-likelihood[2] reflects the generation ability of models. These metrics are evaluated on the test set of PTB, except rPPL and wKL, which are calculated on sentences generated by sampling from these models' prior distribution[3] (sampling a random vector for AE).

Besides, the values of the regularization terms are also included in order to give some indications of the mode-collapse and KL collapse. We list the KL divergence of continuous latent variables (**KL(z)**) and discrete latent variables (**KL(c)**). The weighted variance of parameters (**Var**[4]) and mutual information (**MI**[5]) terms are shown as well.

We first present the ablation study to show the contribution of dispersion term and mutual information term. Results are shown in Tab. 1. First of all, because extra bias is introduced in training objectives (higher mutual information or higher dispersion), the optimization object is no longer the lower bound of log-likelihood. As a result, the NLL will be worse with extra mutual information or dispersion

---

[2] Evaluate by importance sampling (Burda et al., 2015). The number of sampling is 500.

[3] Sample size here is 40,000, the same as PTB training set.

[4] Var is calculated according to Eq. 2 and taking the average over all samples. Variance of multiple parameters are summed.

[5] MI is averaged over multivariate.

| Model	Evaluation Results				Regularization Terms			
	rPPL$^{\downarrow}$	BLEU$^{\uparrow}$	wKL$^{\downarrow}$	NLL$^{\downarrow}$	KL(z)	KL(c)	Var	$\mathcal{L}_{\mathbf{mi}}$
Test Set	-	100.0	**0.14**	-	-	-	-	-
RNN-LM (Mikolov et al., 2010)	-	-	-	101.21	-	-	-	-
AE (Vincent et al., 2010)	730.81	**10.88**	0.58	-	-	-	-	-
VAE (Kingma & Welling, 2013)	686.18	3.12	0.50	≤100.85	5.76	-	-	-
DAE	797.17	3.93	0.58	-	-	-	-	-
DVAE	744.07	1.56	0.55	≤101.07	-	3.87	-	0.17
DI-VAE (Zhao et al., 2018b)	310.29	4.53	0.24	≤108.90	-	24.78	-	1.14
semi-VAE (Kingma et al., 2014)	494.52	2.71	0.43	≤100.67	4.96	1.96	-	0.09
semi-VAE + $\mathcal{L}_{\mathrm{mi}}$	260.28	5.08	0.20	≤107.30	0.045	24.04	-	1.12
GM-VAE	983.50	2.34	0.72	≤**99.44**	3.41	0.00	0.001	0.00
GM-VAE + $\mathcal{L}_{\mathrm{mi}}$	287.07	6.26	0.25	≤103.16	9.13	28.38	13.10	1.30
DGM-VAE	257.68	8.17	0.19	≤104.26	41.48	4.76	787.03	0.22
DGM-VAE + $\mathcal{L}_{\mathrm{mi}}$	**247.37**	**8.67**	**0.18**	≤105.73	25.48	19.73	203.34	0.91

Table 1. Language generation results on PTB. $\beta = 0.8$ for DGM-VAE.
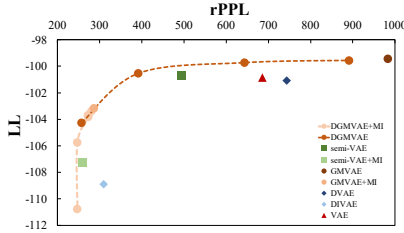
Figure 5. LL and rPPL results for VAE models in PTB.

term. Adding an extra dispersion term helps to alleviate the mode-collapse, according to the higher variance value and mutual information compared with vanilla GM-VAE. Mutual information term helps to increase the information encoded by discrete latent variables, according to higher mutual information. Both terms improve the reconstruction performance and generation quality.

With the presence of both continuous and discrete latent variables, DGM-VAE enjoys its higher model capacity and gives the best reconstruction performance (BLEU), superior to other VAE variants. Although *semi*-VAE also includes discrete and continuous latent variables, it fails to make use of both of them because of the independent hypothesis. As shown in Tab. 1, either discrete or continuous latent variable collapses in *semi*-VAEs. AE could reproduce input sentences well, but it is not a true generative model and fails to generate diverse sentences.

Besides the reconstruction, we also find that DGM-VAE outperforms related work in generating high-quality sentences. rPPL is a powerful metric for measuring the fluency and diversity; DGM-VAE obtains the lowest rPPL. The lowest wKL also shows that word distribution in DGM-VAE generations is most consistent with the training set.

Furthermore, we illustrate the superiority of Gaussian mixture priors embodied by DGM-VAE. The likelihood (LL) and reverse perplexity (rPPL) of DGM-VAE tuning β com-

pared with VAE baselines are shown in Fig. 5. Models, whose results are distributed closer to the upper left region, have better generation quality since it has relatively lower rPPL with the same LL. DGM-VAE is superior to other VAEs, because it has the optimal boundary. Semi-VAE is sub-optimal since it also contains both discrete and continuous latent variables, which is better than VAEs only include discrete or continuous latent variables.

4.4. Interpretable Generation Results

In this section, the interpretable generation results of supervised, unsupervised and conditional text generation are shown. Following Zhao et al. (2018b), we include the experiments of interpretable language generation on DD and dialog generation on SMD.

Unsupervised text generation: DGM-VAE obtains the best performance in interpretability and reconstruction. In this experiment, we show the ability of DGM-VAE to encode the dialog action and emotion information in latent variables unsupervisedly. Because utterances in DD are annotated with Action and Emotion labels, we evaluate the ability of DGM-VAE ($\beta = 0.9$) to unsupervisedly capture these latent attributes on DD. We take the index i with the largest posterior probability $q_\phi(c = i|x)$ as latent action labels. Following Zhao et al. (2018b), we use *homogeneity* as the metric to evaluate the consistency between golden action and emotion labels with labels obtained from DGM-VAE. The number of our labels is 125. Results of homogeneity of action (act) and emotion (em) together with MI term and BLEU are shown in Tab. 2. It shows that DGM-VAE outperforms other VAEs in reconstruction and gives the best homogeneity on both the action and emotion.

Conditional text generation: DGM-VAE gives better dialog generation quality and interpretability. Furthermore, we evaluate the ability of interpretable dialog generation of DGM-VAE based on the structured latent space

	Unsupervised DD			
Model	MI	BLEU$^\uparrow$	act$^\uparrow$	em$^\uparrow$
DI-VAE	1.20	3.05	0.18	0.09
semi-VAE	0.03	4.06	0.02	0.08
semi-VAE $+ \mathcal{L}_{mi}$	1.21	3.69	0.21	0.14
GM-VAE	0.00	2.03	0.08	0.02
GM-VAE $+ \mathcal{L}_{mi}$	1.41	2.96	0.19	0.09
DGM-VAE	0.53	**7.63**	0.11	0.09
DGM-VAE $+ \mathcal{L}_{mi}$	1.32	7.39	**0.23**	**0.16**

Table 2. Results of interpretable language generation on DD. Mutual information (MI), BLEU and homogeneity with actions (act) and emotions (em) are shown. The larger$^\uparrow$, the better.

	Automatic Metrics			
Model	BLEU	Ave.	Ext.	Grd.
DI-VAE	7.06	76.17	43.98	60.92
DGM-VAE $+ \mathcal{L}_{mi}$	**10.16**	**78.93**	**48.14**	**64.87**
	Human Evaluation			
Model	Quality		Consistency	
DI-VAE	2.31		3.08	
DGM-VAE $+ \mathcal{L}_{mi}$	**2.45**		**3.35**	

Table 3. Dialog evaluation results on SMD. Four automatic metrics: BLEU, average (Ave.), extrema (Ext.) and greedy (Grd.) word embedding based similarity are shown. Response quality and consistency within the same c are scored by human.

on SMD dataset. Both automatic evaluation and human evaluation are conducted. BLEU and three word embedding[6] based topic similarity (Serban et al., 2017): Embedding Average, Embedding Extrema and Embedding Greedy (Mitchell & Lapata, 2008; Forgues et al., 2014; Rus & Lintean, 2012) are used to evaluate the quality of responses. In addition, three human evaluators were asked to score the quality (from 0 to 3) of 159 responses generated by DI-VAE and DGM-VAE. Because SMD does not offer human-annotated action labels of dialog utterances, we follow Zhao et al. (2018b) to label dialog actions by human experts for each discrete latent variable c, according to their sampled utterances. Another 3 annotators are asked to evaluate the consistency between the action name and another 5 sampled utterances, showing the interpretability.

Results are shown in Tab. 3. Both automatic and human evaluations show that DGM-VAE obtains better generation quality and interpretability than DI-VAE on SMD. We perform one-tail t-tests on human evaluation scores and find that the superiority of our model is significant in both quality and consistency with p-values no more than 0.05.

We perform case studies to validate the performance of DGM-VAE qualitatively. Some dialog actions with their utterances discovered by DGM-VAE are shown in Tab. 4.

[6]We use GloVe (Pennington et al., 2014) word embeddings of 300 dimension trained on 840B tokens from https://nlp.stanford.edu/projects/glove/.

Action	Inform-route/address
Utterance	There is a Safeway 4 miles away. There are no hospitals within 2 miles. There is Jing Jing and PF Changs.
Action	Request-weather
Utterance	What is the weather today? What is the weather like in the city? What's the weather forecast in New York?

Table 4. Example actions (Act) and corresponding utterances (Utt) discovered by DGM-VAE on SMD. The action name is annotated by experts.

Context	*Sys:* Taking you to Chevron.
Predict	(1-1-3, thanks) Thank you car, let's go there! (1-0-2, request-address) What is the address?
Context	*User:* Make an appointment for the doctor.
Predict	(3-2-4, set-reminder) Setting a reminder for your doctor's appointment on the 12th at 3pm. (3-0-4, request-time) What time would you like to be schedule your doctor's appointment?

Table 5. Dialog cases on SMD, which are generated by sampling different c from policy network. The label of sampled c are listed in parentheses with the annotated action name.

It can be seen that utterances of the same actions could be assigned with the same discrete latent variable c. We also give some dialog cases generated by DGM-VAE in Tab. 5 with their contexts. Given the same context, responses with different actions are generated by sampling different values of discrete latent variables, which shows that DGM-VAE has the ability to generate diverse and interpretable responses. More cases can be found in the supplementary materials.

Supervised text generation: DCM-VAE obtains best performance in multi-label classification accuracy and reconstruction. To show that mode-collapse is a general problem for exponential family mixture VAEs and our proposal is a general solution for it, we conducted supervised text generation experiments using mixture of categorical VAE (CM-VAE). CM-VAE includes mixture of multivariate categorical distribution as priors, which is able to capture the dependency between multiple variables and has been used in multi-label classification in image (Li et al., 2016).

The supervised generation experiment is conducted in DD. Each utterance in DD has emotion and action labels. Multivariate categorical distributions are taken as priors, two of whose components are related to emotion and action through adding cross entropy with golden labels to the objective.

Experimental results of baselines and DCM-VAE ($\beta = 0.4$) are shown in Tab. 6. The classification accuracy of action and emotion is calculated by the corresponding posterior network. DCM-VAE could alleviate the mode-collapse effectively and obtains the best performance in multi-label classification accuracy with slightly worse NLL. It demon-

	Supervised DD			
Model	**NLL**$^\downarrow$	**ACC$_{act}$** $^\uparrow$	**ACC$_{em}$** $^\uparrow$	**Var**
DVAE	$\leq$**48.71**	0.79	0.63	-
CM-VAE	$\leq$52.24	0.79	0.63	0.00
DCM-VAE	$\leq$48.78	**0.80**	**0.73**	139.16

Table 6. Results of supervised interpretable generation on DD. Negative log-likelihood (NLL), classification accuracy (ACC) on emotion (em) and action (act) and the variance of parameters (Var) are shown.

Model	**NLL**$^\downarrow$	**BLEU**$^\uparrow$	$\mathcal{L}_{mi}$	**Var**	**KL**(c)	**KL**(z)
GM-VAE	99.44	2.34	0.00	0.001	0.00	3.41
+ aggressive	99.64	1.47	0.00	0.00	0.00	2.35
+ FB	101.39	6.59	0.19	16.85	5.14	12.52
w/ $\beta = 0.1$	136.08	5.88	0.15	14.89	4.11	49.49
w/ $\beta = 0.5$	103.32	5.92	0.024	1.57	1.43	21.12
w/ $\beta = 0.9$	100.05	1.78	0.00	0.10	0.44	1.52
DGM-VAE	104.26	8.17	0.22	787.03	4.76	41.48

Table 7. Results of DGM-VAE and methods of fixing the posterior collapse. The NLL in the table is a upper-bound estimated by importance sampling.

strates that under supervision, DCM-VAE can inject the interpretable properties in latent variables well. Mode collapse occurs in vanilla CM-VAE, making its performance not better than baseline. DCM-VAE is able to make full use of multiple modes and capture the dependency of variables, which improves the performance of multi-label classification.

4.5. Comparison with KL Collapse Solutions

In this section, we compare our method with previous methods of fixing the KL collapse problem in VAEs. We compare our DGM-VAE with following baselines: 1) β-GM-VAE (Higgins et al., 2017): re-weighting the whole KL term $\mathrm{KL}(q_\phi(z,c)||p_\eta(z,c))$ by β; 2) aggressive update (He et al., 2018): aggressively updating the parameters of posterior network to alleviate posterior collapse; 3) free bits (FB) (Kingma et al., 2016; Li et al., 2019): constrain the values of $\mathrm{KL}(c)$ no larger than λ_c and $\mathrm{KL}(z)$ no larger than λ_z. In our experiment, $\lambda_c = 5$ and $\lambda_z = 10$ are chosen. The same logistic annealing method is applied in KL terms. For β-GM-VAE, we test the performance on a large $\beta = 0.9$, a medium $\beta = 0.5$ and a small $\beta = 0.1$.

Results on PTB are shown in Tab. 7. Among these methods, DGM-VAE is the most effective way to avoid mode-collapse. In the following, we compare with other methods designed to fix KL collapse.

GM-VAE with annealing fails to avoid mode-collapse. Because KL-annealing is not specific for solving the mode-collapse problem, it can prevent the $\mathrm{KL}(z)$ from collapsing to zero but fails to prevent the mode-collapse.

Aggressive updating also fails to fix the mode-collapse. In the early aggressive updating stage, it could keep the $\mathrm{KL}(c)$ term in the non-zero level with $\mathrm{KL}(c) = 8$ but is not well-learned with ELBO = 240. However, mode-collapse still occurs for the subsequent standard VAE training stage. Because no explicit interference on priors is introduced, more updates for posterior network is difficult to guide the priors to be dispersed.

FB can also avoid mode collapse to some extent, but the constraint on the whole $\mathrm{KL}(z)$ term still prefers to make the variance of parameters to be small. Our method could

be combined with the FB by replacing the re-weighting of dispersion term with a maximum value constraint.

β-GM-VAE adjusts the weight of the whole KL term, but it is hard to find a suitable β to maintain a good generative quality while having a non mode-collapse latent space. When β is small (0.1), the NLL is high because weakening the whole KL has a negative effect of under-regularization. When β is relative high (0.5, 0.9), it fails to embed enough information in discrete latent variables as expected.

Our proposed dispersion term is specifically designed for the mode-collapse problem, it is easier to learn a solution with both good generative quality and informative discrete latent variables.

5. Conclusion

In this paper, we theoretical analyze the cause of mode-collapse problem in exponential family mixture VAEs, and propose the dispersed exponential family mixture VAEs to fix it. Experimental results show that our method achieves good performance in interpretable text generation.

References

Bishop, C. M. *Pattern Recognition and Machine Learning*. 2006.

Bowman, S. R., Vilnis, L., Vinyals, O., Dai, A., Jozefowicz, R., and Bengio, S. Generating sentences from a continuous space. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pp. 10–21, 2016.

Brown, P. F., Pietra, V. J. D., Pietra, S. A. D., and Mercer, R. L. The mathematics of statistical machine translation: Parameter estimation. *Computational linguistics*, 19(2): 263–311, 1993.

Burda, Y., Grosse, R., and Salakhutdinov, R. Importance weighted autoencoders. *arXiv preprint arXiv:1509.00519*, 2015.

Chen, X., Duan, Y., Houthooft, R., Schulman, J., Sutskever,

I., and Abbeel, P. Infogan: interpretable representation learning by information maximizing generative adversarial nets. *Neural Information Processing Systems*, pp. 2180–2188, 2016.

Cho, K., van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. Learning phrase representations using rnn encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1724–1734, 2014.

Chung, J., Gulcehre, C., Cho, K., and Bengio, Y. Empirical evaluation of gated recurrent neural networks on sequence modeling. In *NIPS 2014 Workshop on Deep Learning, December 2014*, 2014.

Dilokthanakul, N., Mediano, P. A., Garnelo, M., Lee, M. C., Salimbeni, H., Arulkumaran, K., and Shanahan, M. Deep unsupervised clustering with gaussian mixture variational autoencoders. *arXiv preprint arXiv:1611.02648*, 2016.

Eric, M., Krishnan, L., Charette, F., and Manning, C. D. Key-value retrieval networks for task-oriented dialogue. *Annual Meeting of the Special Interest Group on Discourse and Dialogue*, pp. 37–49, 2017.

Forgues, G., Pineau, J., Larchevêque, J.-M., and Tremblay, R. Bootstrapping dialog systems with word embeddings. In *NIPS, Modern Machine Learning and Natural Language Processing Workshop*, volume 2, 2014.

Gupta, A., Agarwal, A., Singh, P., and Rai, P. A deep generative framework for paraphrase generation. *National Conference on Artificial Intelligence*, 2018.

He, J., Spokoyny, D., Neubig, G., and Berg-Kirkpatrick, T. Lagging inference networks and posterior collapse in variational autoencoders. In *International Conference on Learning Representations*, 2018.

Higgins, I., Matthey, L., Pal, A., Burgess, C. P., Glorot, X., Botvinick, M. M., Mohamed, S., and Lerchner, A. beta-vae: Learning basic visual concepts with a constrained variational framework. *International Conference on Learning Representations*, 2017.

Hu, Z., Yang, Z., Liang, X., Salakhutdinov, R., and Xing, E. P. Toward controlled generation of text. In *International Conference on Machine Learning*, pp. 1587–1596, 2017.

Jang, E., Gu, S., and Poole, B. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

Jiang, Z., Zheng, Y., Tan, H., Tang, B., and Zhou, H. Variational deep embedding: An unsupervised and generative approach to clustering. *International Joint Conference on Artificial Intelligence*, pp. 1965–1972, 2017.

Kingma, D. P. and Welling, M. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*, 2013.

Kingma, D. P., Mohamed, S., Rezende, D. J., and Welling, M. Semi-supervised learning with deep generative models. In *Advances in Neural Information Processing Systems*, pp. 3581–3589, 2014.

Kingma, D. P., Salimans, T., Jozefowicz, R., Chen, X., Sutskever, I., and Welling, M. Improved variational inference with inverse autoregressive flow. In *Advances in neural information processing systems*, pp. 4743–4751, 2016.

Li, B., He, J., Neubig, G., Berg-Kirkpatrick, T., and Yang, Y. A surprisingly effective fix for deep latent variable modeling of text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pp. 3594–3605, 2019.

Li, C., Wang, B., Pavlu, V., and Aslam, J. Conditional bernoulli mixtures for multi-label classification. In *International conference on machine learning*, pp. 2482–2491, 2016.

Li, P., Lam, W., Bing, L., and Wang, Z. Deep recurrent generative decoder for abstractive text summarization. *Empirical Methods in Natural Language Processing*, pp. 2091–2100, 2017a.

Li, Y., Su, H., Shen, X., Li, W., Cao, Z., and Niu, S. Dailydialog: A manually labelled multi-turn dialogue dataset. *International Joint Conference on Natural Language Processing*, 1:986–995, 2017b.

Marcus, M. P., Marcinkiewicz, M. A., and Santorini, B. Building a large annotated corpus of english: the penn treebank. *Computational Linguistics*, 19(2):313–330, 1993.

Merity, S., Keskar, N. S., and Socher, R. Regularizing and optimizing lstm language models. *arXiv preprint arXiv:1708.02182*, 2017.

Miao, Y., Yu, L., and Blunsom, P. Neural variational inference for text processing. *International Conference on Machine Learning*, pp. 1727–1736, 2016.

Mikolov, T., Karafiat, M., Burget, L., Cernocký, J., and Khudanpur, S. Recurrent neural network based language model. *Interspeech*, pp. 1045–1048, 2010.

Mitchell, J. and Lapata, M. Vector-based models of semantic composition. *proceedings of ACL-08: HLT*, pp. 236–244, 2008.

Papineni, K., Roukos, S., Ward, T., and Zhu, W. Bleu: a method for automatic evaluation of machine translation. pp. 311–318, 2002.

Pennington, J., Socher, R., and Manning, C. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 1532–1543, 2014.

Rezende, D. J., Mohamed, S., and Wierstra, D. Stochastic backpropagation and approximate inference in deep generative models. *arXiv preprint arXiv:1401.4082*, 2014.

Rus, V. and Lintean, M. A comparison of greedy and optimal assessment of natural language student input using word-to-word similarity metrics. In *Proceedings of the Seventh Workshop on Building Educational Applications Using NLP*, pp. 157–162. Association for Computational Linguistics, 2012.

Semeniuta, S., Severyn, A., and Barth, E. A hybrid convolutional variational autoencoder for text generation. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 627–637, 2017.

Serban, I. V., Sordoni, A., Bengio, Y., Courville, A., and Pineau, J. Building end-to-end dialogue systems using generative hierarchical neural network models. In *Thirtieth AAAI Conference on Artificial Intelligence*, 2016.

Serban, I. V., Sordoni, A., Lowe, R., Charlin, L., Pineau, J., Courville, A., and Bengio, Y. A hierarchical latent variable encoder-decoder model for generating dialogues. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

Sordoni, A., Bengio, Y., Vahabi, H., Lioma, C., Grue Simonsen, J., and Nie, J.-Y. A hierarchical recurrent encoder-decoder for generative context-aware query suggestion. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, pp. 553–562. ACM, 2015.

Vincent, P., Larochelle, H., Lajoie, I., Bengio, Y., and Manzagol, P. A. Stacked denoising autoencoders: Learning useful representations in a deep network with a local denoising criterion. *Journal of Machine Learning Research*, 11(12):3371–3408, 2010.

Wainwright, M. J., Jordan, M. I., et al. Graphical models, exponential families, and variational inference. *Foundations and Trends® in Machine Learning*, 1(1–2):1–305, 2008.

Wang, W., Gan, Z., Xu, H., Zhang, R., Wang, G., Shen, D., Chen, C., and Carin, L. Topic-guided variational autoencoders for text generation. *arXiv preprint arXiv:1903.07137*, 2019.

Wen, T., Miao, Y., Blunsom, P., and Young, S. J. Latent intention dialogue models. *International Conference on Machine Learning*, pp. 3732–3741, 2017.

Xing, E. P., Jordan, M. I., and Russell, S. A generalized mean field algorithm for variational inference in exponential families. In *Proceedings of the Nineteenth conference on Uncertainty in Artificial Intelligence*, pp. 583–591. Morgan Kaufmann Publishers Inc., 2003.

Xu, J. and Durrett, G. Spherical latent spaces for stable variational autoencoders. *Empirical Methods in Natural Language Processing*, 2018.

Young, S., Gašić, M., Thomson, B., and Williams, J. D. Pomdp-based statistical spoken dialog systems: A review. *Proceedings of the IEEE*, 101(5):1160–1179, 2013.

Zhang, B., Xiong, D., Duan, H., Zhang, M., et al. Variational neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 521–530, 2016.

Zhao, J. J., Kim, Y., Zhang, K., Rush, A. M., and Lecun, Y. Adversarially regularized autoencoders. *International Conference on Machine Learning*, pp. 5897–5906, 2018a.

Zhao, S., Song, J., and Ermon, S. Infovae: Information maximizing variational autoencoders. *arXiv: Learning*, 2017a.

Zhao, T., Zhao, R., and Eskenazi, M. Learning discourse-level diversity for neural dialog models using conditional variational autoencoders. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pp. 654–664, 2017b.

Zhao, T., Lee, K., and Eskenazi, M. Unsupervised discrete sentence representation learning for interpretable neural dialog generation. In *ACL*, 2018b.

Zhou, C. and Neubig, G. Multi-space variational encoder-decoders for semi-supervised labeled sequence transduction. *Meeting of the Association for Computational Linguistics*, 1:310–320, 2017.