Improving Variational Autoencoders with Density Gap-based Regularization

Anonymous Author(s)
Affiliation
Address
email

Abstract

Variational autoencoders (VAEs) are one of the powerful unsupervised learning frameworks in NLP for both latent representation learning and latent-directed generation. The classic optimization goal of VAEs is to maximize the Evidence Lower Bound (ELBo), which consists of a conditional likelihood for generation and a negative Kullback-Leibler (KL) divergence for regularization. In practice, optimizing ELBo often leads the posterior distribution of all samples converge to the same degenerated local optimum, namely *posterior collapse* or *KL vanishing*. There are effective ways proposed to prevent posterior collapse in VAEs, but we observe that they in fact make trade-offs between posterior collapse and *hole problem*, i.e., mismatch between the aggregated (approximate) posterior distribution and the prior distribution. To this end, we propose a new family of VAEs to tackle both two problems through a novel regularization based on the probabilistic *density gap* between the aggregated posterior distribution and the prior distribution. Through experiments on language modeling, latent space visualization and interpolation, we show that our proposed method can solve both problems effectively and thus outperforms the existing methods in terms of latent-guided generation.

1 Introduction

As one of the powerful likelihood-based generative models, variational autoencoders (VAEs) [18, 26] are designed for probabilistic modeling directed by continuous latent variables, which is successfully applied in many NLP tasks, e.g., paraphrase generation [7], controllable text generation [16], neural machine translation [28] and data augmentation [35]. One of the major advantages of VAEs is the flexible latent representation space, which enables easy manipulation of high-level semantics on corresponding representations, e.g., guided sentence generation with interpretable vector operators.

Despite the attractive theoretical strengths, VAEs are observed to suffer from a well-known problem named the *posterior collapse* or the *KL vanishing* problem [18], an optimum state of VAEs when the posterior distribution contains little information about the corresponding datapoint, which is particularly obvious when strong auto-regressive decoders are implemented [38]. Another challenge for VAEs is the *hole problem*, the state when the aggregated (approximate) posterior fails to fit the prior distribution, and thus the inference from prior becomes no longer suitable to describe the global data distribution [27], which can lead to poor generation quality in VAEs [1, 21].

In this work, we take systematic experiments on VAEs for text generation to study the *posterior collapse* and the *hole problem* in existing methods. We demonstrate that VAEs with specific network structures [9, 37] or modified training strategies [5, 11] have limited effect on solving posterior collapse, while VAEs with hard restrictions [8, 33, 38] or weakened KL regularization [14, 17] on the posterior can solve posterior collapse effectively at the expense of introducing hole problem, as illustrated in Figure 1.

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

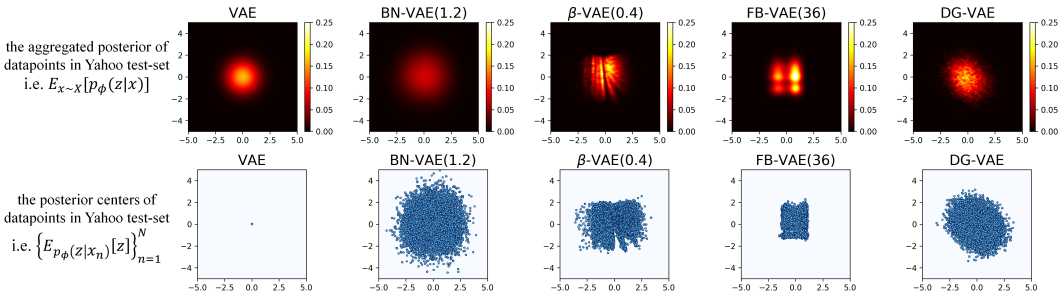

Figure 1: The visualization of aggregated posterior (the first line) and posterior centers (the second line) for models on Yahoo test-set. The vanilla VAEs suffer from *posterior collapse*, i.e., the posterior centers collapse to the same position. Meanwhile, BN-VAEs [38], β-VAEs [14] and FB-VAEs [17] can solve posterior collapse effectively at the cost of bringing *hole problem*, i.e., mismatch between the aggregated posterior and the prior. Our proposed DG-VAE intends to solve both problems through a novel regularization based on *density gap*. Complete illustrations are shown in Appendix G.

On that basis, we hypothesize those two problems stem from the conflict between the KL regularization in ELBo and the function definition of the prior. So we propose a novel regularization to substitute the KL regularization in ELBo for VAEs, which is based on the *density gap* between the aggregated posterior and the prior. We provide theoretical proof that our method in essence maximizes the ELBo as well as the mutual information between the input and the latent variable.

In terms of Gaussian distribution-based VAEs, we further propose the corresponding marginal regularization on each dimension respectively, and we prove it in essence maximizes the ELBo as well as the sum of mutual information between the input and the latent variable on all dimensions.

To validate our methods in practice, we take experiments on language modeling, latent-guided generation and latent space visualization. We demonstrate that our methods form latent spaces both active and consistent with the prior, and thus generate smoother sentences from latent interpolation.

2 Background and Related Work

2.1 VAEs and ELBo

VAEs are proposed to perform efficient inference and learning in directed probabilistic models, where the random generation process consists of two steps: (1) sample a latent value $\mathbf{z}$ from the prior $p_{\boldsymbol{\theta}}(\mathbf{z})$; (2) generate a datapoint $\mathbf{y}$ from the conditional distribution $p_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{z})$. As the true posterior $p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{y})$ is intractable, a recognition model $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$ is introduced to approximate the true posterior [18].

Specifically, VAEs represent an input $\mathbf{x}$ as a latent distribution $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$, from which latent variables are sampled to direct the generation of the desired output $\mathbf{y}$, which is commonly the same sentence or image to $\mathbf{x}$ for unsupervised learning. As the optimization goal of VAEs, the Evidence Lower Bound (ELBo) is composed of a Kullback-Leibler (KL) divergence for regularization on posterior and a log likelihood for reconstruction conditioned on posterior. The ELBo in fact forms a lower bound on the marginal likelihood given prior latent variables [18], as illustrated in Eq. 1.

$$\begin{aligned}\mathcal{L}_{ELBo}(\boldsymbol{\theta},\boldsymbol{\phi};\mathbf{x},\mathbf{y}) &= \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}[\log p_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{z})] - D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})\|p_{\boldsymbol{\theta}}(\mathbf{z})) \\ &\leq \log p_{\boldsymbol{\theta}}(\mathbf{y}) = \log \int p_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{z})p_{\boldsymbol{\theta}}(\mathbf{z})d\mathbf{z}\end{aligned} \tag{1}$$

2.2 Decomposition of KL Divergence in ELBo

The *aggregated (approximate) posterior* $q_{\boldsymbol{\phi}}(\mathbf{z})$ refers to the expectation of approximate posterior distribution on data distribution, as defined in Eq. 2, where the distribution of input data $\mathbf{x}$ is represented by the discrete distribution of datapoints in the dataset $X = \{x_n\}_{n=1}^N, p(\mathbf{x} = x_n) = \frac{1}{N}$.

$$q_{\boldsymbol{\phi}}(\mathbf{z}) = \mathbb{E}_{\mathbf{x}}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})) = \frac{1}{N}\sum_{n=1}^{N} q_{\boldsymbol{\phi}}(\mathbf{z}|x_n) \tag{2}$$

On that basis, Hoffman et al. [15] treat the index n as a random variable, and decompose the KL term in ELBo into the sum of KL divergence between $q_{\boldsymbol{\phi}}(\mathbf{z})$ and $p_{\boldsymbol{\theta}}(\mathbf{z})$ and the mutual information of $\mathbf{z}$ and n in their joint distribution $q_{\boldsymbol{\phi}}(n, \mathbf{z})$, as stated in Eq. 3, which provides theoretical support for our proposal.

$$\begin{aligned} \frac{1}{N}\sum_{n=1}^{N} D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}|x_n)\|p_{\boldsymbol{\theta}}(\mathbf{z})) &= D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z})\|p_{\boldsymbol{\theta}}(\mathbf{z})) + \mathbb{I}_{q_{\boldsymbol{\phi}}(n,\mathbf{z})}[n, \mathbf{z}] \\ \text{where } \mathbb{I}_{q_{\boldsymbol{\phi}}(n,\mathbf{z})}[n, \mathbf{z}] &= \mathbb{E}_{q_{\boldsymbol{\phi}}(n,\mathbf{z})}[\log \frac{q_{\boldsymbol{\phi}}(n, \mathbf{z})}{q_{\boldsymbol{\phi}}(n) q_{\boldsymbol{\phi}}(\mathbf{z})}] \\ \text{where } q_{\boldsymbol{\phi}}(n, \mathbf{z}) &= q_{\boldsymbol{\phi}}(n) q_{\boldsymbol{\phi}}(\mathbf{z}|n) = \frac{1}{N} q_{\boldsymbol{\phi}}(\mathbf{z}|x_n) \end{aligned} \tag{3}$$

2.3 Posterior Collapse

Intuitively, the KL divergence in ELBo (i.e., $D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})\|p_{\boldsymbol{\theta}}(\mathbf{z}))$) encourages the approximate posterior of every single datapoint to be close to the the prior [18]. This intends to ensure the prior can depict the latent distribution over data distribution, but it can also lead to posterior collapse when $D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})\|p_{\boldsymbol{\theta}}(\mathbf{z}))$ has much stronger force on $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})$ than $\mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}[\log p_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{z})]$ does, which leads that $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}) \approx p_{\boldsymbol{\theta}}(\mathbf{z})$ for nearly every $\mathbf{x}$ and thus the model performs posterior collapse.

Early works to solve posterior collapse attribute it to the difficulty in optimizing ELBo, so their methods mainly focus on the training strategies [5, 11]. Some works put emphasis on semantic learning of the posterior through specific model structures [9, 37]. Instead of Gaussian distribution, vMF-VAE [8, 33] adopts von Mises-Fisher distribution for latent variables, which restricts the posterior latent in a hyperspherical space and forms a constant KL divergence. BN-VAE [38] restricts the posterior distribution through a batch normalization layer with a fixed scale γ, so as to guarantee a positive lower bound of the KL divergence. β-VAE [14] directly changes the weight (denoted as β) of KL term inside ELBo, while free-bits [17] changes the KL term inside ELBo to a hinge loss term: $\sum_i^{Dim} \max(\frac{\lambda_{KL}}{Dim}, D_{KL}(q_{\boldsymbol{\phi}}(z_i|\mathbf{x})\|p_{\boldsymbol{\theta}}(z_i)))$.

In contrast, we hypothesize the posterior collapse is due to the conflict between the KL regularization in ELBo and the function definition of the prior, and tackle it through replacing the KL regularization in ELBo with a novel regularization on the aggregated posterior.

2.4 Hole Problem

Formally, the hole problem refers to the phenomenon that the aggregated posterior distribution $q_{\boldsymbol{\phi}}(\mathbf{z})$ fails to fit into the prior distribution $p_{\boldsymbol{\theta}}(\mathbf{z})$, which is an unwanted status as the prior distribution is no longer suitable to represent the data distribution in latent space. Inferences located in the holes (i.e., areas with mismatch between density in $q_{\boldsymbol{\phi}}(\mathbf{z})$ and $p_{\boldsymbol{\theta}}(\mathbf{z})$) are observed to generate images that are obscure and corrupted [1], or sentences with problematic word matching and content against commonsense [21].

The hole problem of VAEs for image generation is observed in several studies, commonly ascribed to the limited expressivity of the prior and tackled by increasing the flexibility of the prior via hierarchical priors [19], auto-regressive models [13], normalizing flows [32], resampled priors [3] and energy-based models [1]. In contrast, we observe that the vanilla VAEs (with static prior) for text generation has no hole problem, but it arises when existing methods are applied to solve posterior collapse. So our work is targeted at solving posterior collapse and avoiding hole problem at the same time, for VAEs with static prior.

2.5 Regularization on Aggregated Posterior

As the approximate posterior is introduced to approximate the true posterior, i.e., $q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}) \approx p_{\boldsymbol{\theta}}(\mathbf{z}|\mathbf{x})$, the aggregated posterior should be close to the prior as a result, i.e., $q_{\boldsymbol{\phi}}(\mathbf{z}) \approx p_{\boldsymbol{\theta}}(\mathbf{z})$. From this point of view, several works are proposed to replace KL regularization (on posterior for single datapoint) in ELBo with regularization on the aggregated posterior, which can be summarized as Eq. 4, where D is the divergence (or discrepancy) between two distributions.

$$\mathcal{L}_D(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}, \mathbf{y}) = \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}[\log p_{\boldsymbol{\theta}}(\mathbf{y}|\mathbf{z})] - D(q_{\boldsymbol{\phi}}(\mathbf{z})\|p_{\boldsymbol{\theta}}(\mathbf{z})) \tag{4}$$

Among them, Adversarial Auto-Encoder (AAE) [23] adopts the Generative Adversarial Network (GAN) [12] framework to regularize the aggregated posterior through Jensen–Shannon divergence D_{JS}; Wasserstein Auto-Encoder (WAE) [30, 2] regularizes the aggregated posterior through minimizing Maximum Mean Discrepancy (MMD); Implicit VAE with Mutual Information regularization (iVAE$_{MI}$) regularizes the aggregated posterior through the dual form of KL divergence D_{KL} on the basis of Implicit VAE (iVAE) [10]. Those methods have a same weakness that their approximations of the divergence between two continuous distributions are depicted by merely sampling sets from the distributions, which involves noise from random sampling and can hardly be zero even for the same distribution. In contrast, our method approximates the divergence between two continuous distributions in the perspective of their mismatch in PDFs, which we quantify through *density gap* that can be zero if and only if they are the same, which we describe in section 3.

We validate our method in comparison with those methods through experiments on a synthetic dataset, the details and results of which are demonstrated in Appendix A.

3 Methodology

Density Gap-based Discrepancy One of the straight manifestations of *holes* in latent space is the mismatch of probabilistic density between $q_{\boldsymbol{\phi}}(\mathbf{z})$ and $p_{\boldsymbol{\theta}}(\mathbf{z})$, we quantify this mismatch at a specific position z in the latent space through $DG(\boldsymbol{\theta}, \boldsymbol{\phi}; z)$, which we refer to as *Density Gap*. Here we only consider $z \in \{z | q_{\boldsymbol{\phi}}(z) > 0\}$. We assume $q_{\boldsymbol{\phi}}(z)$ and $p_{\boldsymbol{\theta}}(z)$ are differentiable and $p_{\boldsymbol{\theta}}(z) > 0$.

$$DG(\boldsymbol{\theta}, \boldsymbol{\phi}; z) = \log \frac{q_{\boldsymbol{\phi}}(z)}{p_{\boldsymbol{\theta}}(z)} = \log \frac{\frac{1}{N}\sum_{n=1}^{N} q_{\boldsymbol{\phi}}(z|x_n)}{p_{\boldsymbol{\theta}}(z)} \tag{5}$$

It can be inferred that the expectation of $DG(\boldsymbol{\theta}, \boldsymbol{\phi}; z)$ on $q_{\boldsymbol{\phi}}(\mathbf{z})$ equals to the KL divergence between $q_{\boldsymbol{\phi}}(\mathbf{z})$ and $p_{\boldsymbol{\theta}}(\mathbf{z})$, as illustrated in Eq. 6, which is a strict divergence, i.e., $\mathbb{E}_{z\sim q_{\boldsymbol{\phi}}(\mathbf{z})}[DG(\boldsymbol{\theta}, \boldsymbol{\phi}; z)] = 0$ iff $q_{\boldsymbol{\phi}}(\mathbf{z}) = p_{\boldsymbol{\theta}}(\mathbf{z})$.

$$\mathbb{E}_{z\sim q_{\boldsymbol{\phi}}(\mathbf{z})}[DG(\boldsymbol{\theta}, \boldsymbol{\phi}; z)] = \mathbb{E}_{z\sim q_{\boldsymbol{\phi}}(\mathbf{z})}[\log \frac{q_{\boldsymbol{\phi}}(z)}{p_{\boldsymbol{\theta}}(z)}] = D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}) \| p_{\boldsymbol{\theta}}(\mathbf{z})) \geq 0 \tag{6}$$

So, we can approximate and optimize $D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}) \| p_{\boldsymbol{\theta}}(\mathbf{z}))$ via Monte Carlo, as illustrated in Eq. 7, where $z_s \overset{idd}{\sim} q_{\boldsymbol{\phi}}(\mathbf{z})$ denotes the sth random sample from the aggregated posterior distribution.

$$D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}) \| p_{\boldsymbol{\theta}}(\mathbf{z})) \approx \frac{1}{S}\sum_{s=1}^{S} DG(\boldsymbol{\theta}, \boldsymbol{\phi}; z_s) \tag{7}$$

It should be noticed that $D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}) \| p_{\boldsymbol{\theta}}(\mathbf{z}))$ approximated by this is an *overall* divergence, as it considers the posterior distribution of all datapoints as a whole, instead of averaging $D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x}) \| p_{\boldsymbol{\theta}}(\mathbf{z}))$ across all datapoints as ELBo does.

On that basis, we can implement $\mathcal{L}_D(\boldsymbol{\theta}, \boldsymbol{\phi}; \mathbf{x}, \mathbf{y})$ with $D = D_{KL}$, which is equivalent to replacing the KL term in ELBo with $D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}) \| p_{\boldsymbol{\theta}}(\mathbf{z}))$ approximated by Eq. 7. According to the decomposition (illustrated in Eq. 3) of KL term in ELBo given by Hoffman et al. [15], maximizing $\mathcal{L}_{D_{KL}}(\boldsymbol{\theta}, \boldsymbol{\phi}; x_n, y_n)$ on the whole dataset is equivalent to maximizing ELBo as well as $\mathbb{I}_{q_{\boldsymbol{\phi}}(n,\mathbf{z})}[n, \mathbf{z}]$, the mutual information of $\mathbf{z}$ and n in their joint distribution $q_{\boldsymbol{\phi}}(n, \mathbf{z})$, as illustrated in Eq. 8.

$$\begin{aligned} \frac{1}{N}\sum_{n=1}^{N}[\mathcal{L}_{D_{KL}}(\boldsymbol{\theta}, \boldsymbol{\phi}; x_n, y_n)] &= \frac{1}{N}\sum_{n=1}^{N}[\mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|x_n)}[\log p_{\boldsymbol{\theta}}(y_n|\mathbf{z})]] - D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}) \| p_{\boldsymbol{\theta}}(\mathbf{z})) \\ &= \frac{1}{N}\sum_{n=1}^{N}[\mathcal{L}_{ELBo}(\boldsymbol{\theta}, \boldsymbol{\phi}; x_n, y_n)] + \mathbb{I}_{q_{\boldsymbol{\phi}}(n,\mathbf{z})}[n, \mathbf{z}] \end{aligned} \tag{8}$$

Optimization on Mini-Batch Theoretically attractive as it is, maximizing $\mathcal{L}_{D_{KL}}(\boldsymbol{\theta}, \boldsymbol{\phi}; x_n, y_n)$ approximated by $DG(\boldsymbol{\theta}, \boldsymbol{\phi}; z_s)$ is undesirable for training VAEs on large datasets, because the probabilistic density of $q_{\boldsymbol{\phi}}(\mathbf{z})$ at z needs computation across the whole dataset and is changing in

every training step. In practice, training deep networks such like VAEs commonly adopts mini-batch gradient descent, where only a small subset of the dataset is used for calculating gradients and updating parameters in each iteration step.

So, a practicable way is to aggregate the posterior of datapoints inside a mini-batch $B = \{x_n\}_{n=1}^{|B|}$, as stated in Eq. 9, where $z_{n,m}$ is the mth sample from the posterior of datapoint x_n. Here stratified sampling (i.e., sampling $S = |B| \times M$ samples from $q_{\boldsymbol{\phi},B}(z)$ through sampling M samples from $q_{\boldsymbol{\phi}}(\mathbf{z}|x_n)$ for each datapoint $x_n \in B$) is used to ensure a steady Monte Carlo approximation, and the reparameterization trick [18] is applied to ensure a differentiable output.

$$\begin{aligned} DG(\boldsymbol{\theta}, \boldsymbol{\phi}, B; z) &= \log \frac{q_{\boldsymbol{\phi},B}(z)}{p_{\boldsymbol{\theta}}(z)} = \log \frac{\frac{1}{|B|}\sum_{n=1}^{|B|} q_{\boldsymbol{\phi}}(z|x_n)}{p_{\boldsymbol{\theta}}(z)} \\ D_{KL}(q_{\boldsymbol{\phi},B}(\mathbf{z})\|p_{\boldsymbol{\theta}}(\mathbf{z})) &\approx \frac{1}{|B|}\sum_{n=1}^{|B|}\frac{1}{M}\sum_{m=1}^{M} DG(\boldsymbol{\theta}, \boldsymbol{\phi}, B; z_{n,m}), \text{where } z_{n,m} \overset{idd}{\sim} q_{\boldsymbol{\phi}}(\mathbf{z}|x_n) \end{aligned} \tag{9}$$

Through this approximation, we can implement $\mathcal{L}_{D_{KL}}(\boldsymbol{\theta}, \boldsymbol{\phi}, B; x_n, y_n)$ that regularizes $q_{\boldsymbol{\phi},B}(\mathbf{z})$ towards $p_{\boldsymbol{\theta}}(\mathbf{z})$ for a mini-batch B, as stated in Eq. 10.

$$\begin{aligned} \frac{1}{|B|}\sum_{n=1}^{|B|}[\mathcal{L}_{D_{KL}}(\boldsymbol{\theta}, \boldsymbol{\phi}, B; x_n, y_n)] &= \frac{1}{|B|}\sum_{n=1}^{|B|}[\mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|x_n)}[\log p_{\boldsymbol{\theta}}(y_n|\mathbf{z})]] - D_{KL}(q_{\boldsymbol{\phi},B}(\mathbf{z})\|p_{\boldsymbol{\theta}}(\mathbf{z})) \\ &= \frac{1}{|B|}\sum_{n=1}^{|B|}[\mathcal{L}_{ELBo}(\boldsymbol{\theta}, \boldsymbol{\phi}; x_n, y_n)] + \mathbb{I}_{q_{\boldsymbol{\phi}}(n,\mathbf{z})}[n, \mathbf{z}] \end{aligned} \tag{10}$$

It should be noticed that the mutual information term $\mathbb{I}_{q_{\boldsymbol{\phi}}(n,\mathbf{z})}[n, \mathbf{z}]$ is different in Eq. 8 (on the whole dataset) and Eq. 10 (on a mini-batch), because the range of discrete variable n is from 1 to N in Eq. 8 but 1 to $|B|$ in Eq. 10. Consequently, $\mathbb{I}_{q_{\boldsymbol{\phi}}(n,\mathbf{z})}[n, \mathbf{z}]$ has a upper bound of $H(n) = \log N$ in Eq. 8 but $H(n) = \log |B|$ in Eq. 10. In other words, maximizing $\mathbb{I}_{q_{\boldsymbol{\phi}}(n,\mathbf{z})}[n, \mathbf{z}]$ in Eq. 8 intends to distinguish $\mathbf{z}$ of x_n from that of $N-1$ other datapoints while it is limited to $|B|-1$ other datapoints in Eq. 10.

Marginal Regularization for More Mutual Information As described above, approximating and optimizing $D_{KL}(q_{\boldsymbol{\phi},B}(\mathbf{z})\|p_{\boldsymbol{\theta}}(\mathbf{z}))$ is practicable but has limited effect. Empirically, Gaussian distribution-based VAEs trained by Eq. 10 still have limited active units, which means the encoded latent variable $\mathbf{z}$ still collapses to the prior on most dimensions, where it provides little information.

To activate $\mathbf{z}$ on all dimensions, we propose to regularize $q_{\boldsymbol{\phi},B}(\mathbf{z})$ towards $p_{\boldsymbol{\theta}}(\mathbf{z})$ on each dimension respectively, i.e., regularize the marginal distribution on each dimension, as illustrated in Eq. 11, where $z_i \in R$ is the ith component of $z \in R^{Dim}$ and the corresponding probability density functions are of the marginal distributions on the ith dimension; $z_{n,m,i}$ is the ith component of the mth sample from the posterior of datapoint x_n.

$$\begin{aligned} DG_{mrg}(\boldsymbol{\theta}, \boldsymbol{\phi}, B; z_i) &= \log \frac{q_{\boldsymbol{\phi},B}(z_i)}{p_{\boldsymbol{\theta}}(z_i)} = \log \frac{\frac{1}{|B|}\sum_{n=1}^{|B|} q_{\boldsymbol{\phi}}(z_i|x_n)}{p_{\boldsymbol{\theta}}(z_i)} \\ D_{KL,mrg}(q_{\boldsymbol{\phi},B}(\mathbf{z})\|p_{\boldsymbol{\theta}}(\mathbf{z})) &= \sum_{i=1}^{Dim} D_{KL}(q_{\boldsymbol{\phi},B}(\mathbf{z}_i)\|p_{\boldsymbol{\theta}}(\mathbf{z}_i)) \\ &\approx \sum_{i=1}^{Dim}\frac{1}{|B|}\sum_{n=1}^{|B|}\frac{1}{M}\sum_{m=1}^{M} DG_{mrg}(\boldsymbol{\theta}, \boldsymbol{\phi}, B; z_{n,m,i}) \end{aligned} \tag{11}$$

In Gaussian distribution-based VAEs, the marginal distributions of $\mathbf{z}$ on different dimensions are independent, i.e., $q_{\boldsymbol{\phi}}(\mathbf{z}|x_n) = \prod_i^{Dim} q_{\boldsymbol{\phi}}(\mathbf{z}_i|x_n)$ and $p_{\boldsymbol{\theta}}(\mathbf{z}) = \prod_i^{Dim} p_{\boldsymbol{\theta}}(\mathbf{z}_i)$, so their KL divergence can be decomposed as $D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}|x_n)\|p_{\boldsymbol{\theta}}(\mathbf{z})) = \sum_i^{Dim} D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}_i|x_n)\|p_{\boldsymbol{\theta}}(\mathbf{z}_i))$. Thus, we can

infer the decomposition of $D_{KL,mrg}(q_{\boldsymbol{\phi},B}(\mathbf{z})\|p_{\boldsymbol{\theta}}(\mathbf{z}))$ through Eq. 12.

$$\begin{aligned} D_{KL,mrg}(q_{\boldsymbol{\phi},B}(\mathbf{z})\|p_{\boldsymbol{\theta}}(\mathbf{z})) &= \sum_{i=1}^{Dim} D_{KL}(q_{\boldsymbol{\phi},B}(\mathbf{z}_i)\|p_{\boldsymbol{\theta}}(\mathbf{z}_i)) \\ &= \sum_{i=1}^{Dim} [\frac{1}{|B|}\sum_{n}^{|B|}[D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}_i|x_n)\|p_{\boldsymbol{\theta}}(\mathbf{z}_i))] - \mathbb{I}_{q_{\boldsymbol{\phi}}(n,\mathbf{z}_i)}[n,\mathbf{z}_i]] \\ &= \frac{1}{|B|}\sum_{n}^{|B|}[D_{KL}(q_{\boldsymbol{\phi}}(\mathbf{z}|x_n)\|p_{\boldsymbol{\theta}}(\mathbf{z}))] - \sum_{i=1}^{Dim}[\mathbb{I}_{q_{\boldsymbol{\phi}}(n,\mathbf{z}_i)}[n,\mathbf{z}_i]] \end{aligned} \tag{12}$$

So, maximizing $\mathcal{L}_{D_{KL,mrg}}(\boldsymbol{\theta},\boldsymbol{\phi},B;x_n,y_n)$ derived from Eq. 11 is equivalent to maximizing ELBo as well as the mutual information of n and $\mathbf{z}_i$ for each dimension i respectively, as stated in Eq. 13. In other words, it intends to distinguish $\mathbf{z}_i$ of x_n from that of $|B|-1$ other datapoints for each dimension i respectively. We refer to our models based on this Density Gap-based regularization as DG-VAEs.

$$\begin{aligned} \frac{1}{|B|}\sum_{n=1}^{|B|}[\mathcal{L}_{D_{KL,mrg}}(\boldsymbol{\theta},\boldsymbol{\phi},B;x_n,y_n)] &= \frac{1}{|B|}\sum_{n=1}^{|B|}[\mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|x_n)}[\log p_{\boldsymbol{\theta}}(y_n|\mathbf{z})]] - D_{KL,mrg}(q_{\boldsymbol{\phi},B}(\mathbf{z})\|p_{\boldsymbol{\theta}}(\mathbf{z})) \\ &= \frac{1}{|B|}\sum_{n=1}^{|B|}[\mathcal{L}_{ELBo}(\boldsymbol{\theta},\boldsymbol{\phi};x_n,y_n)] + \sum_{i=1}^{Dim}[\mathbb{I}_{q_{\boldsymbol{\phi}}(n,\mathbf{z}_i)}[n,\mathbf{z}_i]] \end{aligned} \tag{13}$$

Aggregation Size for Ablation As discussed above, the size of mini-batch $|B|$ sets a upper bound of the mutual information term $\mathbb{I}_{q_{\boldsymbol{\phi}}(n,\mathbf{z})}[n,\mathbf{z}]$ (or $\mathbb{I}_{q_{\boldsymbol{\phi}}(n,\mathbf{z}_i)}[n,\mathbf{z}_i]$). To validate this impact, we further extend DG-VAEs through dividing the mini-batch into non-overlapping subsets $B = \bigcup_{i=1}^{C} b_i$, s.t. $b_j \cap b_i = \emptyset$ iff $i \neq j$, calculating and optimizing the KL divergence over each subset, i.e., $\frac{1}{C}\sum_{i=1}^{C}\frac{1}{|b_i|}\sum_{j=1}^{|b_i|}[\mathcal{L}_{D_{KL,mrg}}(\boldsymbol{\theta},\boldsymbol{\phi},b_i;x_n,y_n)]$, where C denotes the number of subsets and $|b_i| = \frac{|B|}{C}$ is the size of those subsets, which we refer to as the *aggregation size* and denote as $|b|$ for simplification. It can be inferred that the DG-VAE with $|b| = 1$ is equivalent to the vanilla VAE trained by ELBo, except that it approximates the KL term through Monte Carlo instead of integration.

Extension to von Mises-Fisher Distribution-based VAEs Besides the commonly used Gaussian distribution-based VAEs, we also consider von Mises-Fisher (vMF) distribution-based VAEs. As the decomposition $q_{\boldsymbol{\phi}}(\mathbf{z}) = \prod_{i}^{Dim} q_{\boldsymbol{\phi}}(\mathbf{z}_i)$ is not established for latent variables following vMF distribution (i.e., $\mathbf{z} \sim vMF(\boldsymbol{\mu},\kappa)$), marginal regularization for vMF-VAEs may not be interpretable, so we only implement Eq. 10 in vMF-VAEs. We refer to those extensions as DG-vMF-VAEs.

4 Experiments

4.1 Experimental Setup

Datasets We consider four public available datasets commonly used for VAE-based language modeling task in our experiments: Yelp [34], Yahoo [34, 36], a downsampled version of Yelp [29] (we denote this as Short-Yelp) and a downsampled version of SNLI [4, 20]. The statistics of those datasets are illustrated in Table 1. It can be viewed that Yelp and Yahoo contain long sentences while Short-Yelp and SNLI contain short sentences.

Baselines We consider a wide range of VAEs for solving posterior collapse in text generation, where the hyperparameters are set according to Zhu et al. [38]:

- VAEs with modified training strategies (i.e., KL annealing): VAE with linear KL annealing in the first 10 epochs (default) [5]; VAE with linear KL annealing for 10 epochs at the start of every 20 epochs (cyclic-VAE) [11];
- VAEs with specific model structures: VAE with additional Bag-of-Words loss (bow-VAE) [37], and VAE with skip connection from the latent variable $\mathbf{z}$ to the vocabulary classifier for generation (skip-VAE) [9];

Table 1: Statistics of sentences in the datasets

Dataset	Train	Valid	Test	Vocab size	Length (avg $\pm$ std)
Yelp	100,000	10,000	10,000	19997	98.01 $\pm$ 48.86
Yahoo	100,000	10,000	10,000	20001	80.76 $\pm$ 46.21
Short-Yelp	100,000	10,000	10,000	8411	10.96 $\pm$ 3.60
SNLI	100,000	10,000	10,000	9990	11.73 $\pm$ 4.33

Table 2: Results of Language Modeling on Yahoo dataset. We bold up $MI(\boldsymbol{\phi}) \geq 9.0$, $AU(\boldsymbol{\phi}) \geq 30$, $CU(\boldsymbol{\phi}) \geq 30$, the highest $priorLL(\boldsymbol{\theta})$ and $postLL(\boldsymbol{\theta}, \boldsymbol{\phi})$ for the same methods.

Models	$priorLL(\boldsymbol{\theta})$	$postLL(\boldsymbol{\theta}, \boldsymbol{\phi})$	$KL(\boldsymbol{\phi})$	$MI(\boldsymbol{\phi})$	$AU(\boldsymbol{\phi})$	$CU(\boldsymbol{\phi})$
VAE (default)	-330.7	-330.7	0.0	0.0	0	**32**
cyclic-VAE	-329.8	-328.9	1.1	1.0	2	**31**
bow-VAE	-330.5	-330.5	0.0	0.0	0	**32**
skip-VAE	-330.1	-325.2	5.0	4.3	8	**31**
δ-VAE(0.15)	-330.5	-330.6	4.8	0.0	0	0
BN-VAE(0.6)	**-327.6**	-321.1	6.6	5.9	**32**	**32**
BN-VAE(1.2)	-330.9	-310.1	26.2	**9.2**	**32**	0
BN-VAE(1.8)	-343.5	**-308.6**	51.3	**9.2**	**32**	0
FB-VAE(4)	-329.8	-328.4	3.9	1.8	**32**	**32**
FB-VAE(16)	**-325.7**	-320.8	16.1	8.5	**32**	8
FB-VAE(49)	-344.6	**-296.1**	50.0	**9.2**	**32**	0
β-VAE(0.4)	**-330.8**	-324.8	7.0	6.7	3	**31**
β-VAE(0.2)	-338.6	-310.3	30.1	**9.2**	22	25
β-VAE(0.1)	-369.9	**-289.6**	83.7	**9.2**	**32**	0
DG-VAE ($\lvert b\rvert = 1$)	-330.7	-330.7	0.0	0.0	0	**32**
DG-VAE ($\lvert b\rvert = 4$)	**-330.4**	-318.3	14.3	**9.1**	11	**32**
DG-VAE ($\lvert b\rvert = 32$)	-355.4	-294.1	65.2	**9.1**	**32**	**32**
DG-VAE (default)	-358.0	**-290.8**	70.8	**9.1**	**32**	**32**

- VAEs with hard restrictions on the posterior distribution: δ-VAE with the committed rate $\delta = 0.15$ [25]; BN-VAEs with the scale of BN layer $\gamma \in \{0.6, 0.7, 0.9, 1.2, 1.5, 1.8\}$ [38]; vMF-VAEs with the distribution's concentration $\kappa \in \{13, 25, 50, 100, 200\}$ [8, 33];
- VAEs with weakened KL regularization: FB-VAEs (free-bits) with the target KL[1] $\lambda_{KL} \in \{4, 9, 16, 25, 36, 49\}$ [17]; β-VAEs with the weight of KL term $\beta \in \{0.0, 0.1, 0.2, 0.4, 0.8\}$ [14].

Configurations We completely follow Zhu et al. [38] in models' backbone structures, data preprocessing and training procedure, which we describe in details in Appendix B.

4.2 Language Modeling

We evaluate the performance of our methods and the baselines on language modeling,[2] where the following metrics are reported: the prior log likelihood $priorLL(\boldsymbol{\theta})$ and the posterior log likelihood $postLL(\boldsymbol{\theta}, \boldsymbol{\phi})$ for generation quality; the KL term in ELBo $KL(\boldsymbol{\phi})$, the mutual information $MI(\boldsymbol{\phi})$ of $\mathbf{z}$ and n and the number of active units $AU(\boldsymbol{\phi})$ [6] for posterior collapse; the number of consistent units $CU(\boldsymbol{\phi})$ (we propose) for hole problem. The corresponding expressions and explanations are in the Appendix C.

We illustrate part of the results on Yahoo in Table 2 and all results on all datasets in Appendix D. It can be observed that: (1) models with modified training strategies or specific model structures can alleviate the problem of posterior collapse but has limited effect according to $MI(\boldsymbol{\phi})$ and $AU(\boldsymbol{\phi})$; (2) models with hard restrictions or weakened KL regularization on the posterior can solve posterior collapse better through harder restrictions or further weakening according to the increase of $KL(\boldsymbol{\phi})$,

[1] We also refer to this as its total threshold in hinge loss.

[2] In our experiment, the input and output of the model are the same sentence, i.e., $\mathbf{x} = \mathbf{y}, \mathbf{X} = \mathbf{Y}$

$MI(\boldsymbol{\phi})$ and $AU(\boldsymbol{\phi})$, but the decrease of $CU(\boldsymbol{\phi})$ indicates that their posterior latent spaces tend to be increasingly inconsistent with that of prior; (3) in contrast, our proposed DG-VAE has the similar performance to the vanilla VAE when $|b| = 1$, and with the increase of $|b|$, it can solve the posterior collapse effectively and avoid the hole problem at the same time.[3]

It can also be viewed that with the increase of $MI(\boldsymbol{\phi})$ or $KL(\boldsymbol{\phi})$, $postLL(\boldsymbol{\theta}, \boldsymbol{\phi})$ tends to increase while $priorLL(\boldsymbol{\theta})$ tends to decrease, as the decoder $\boldsymbol{\theta}$ becomes more dependent on the encoder $\boldsymbol{\phi}$. We further plot the curves of $priorLL(\boldsymbol{\theta})$ and $postLL(\boldsymbol{\theta}, \boldsymbol{\phi})$ for models with different hyperparameters in Figure 2, where we can observe that DG-VAEs make better trade-offs than BN-VAEs and β-VAEs do on short datasets and perform competitively to BN-VAEs and FB-VAEs on long datasets.

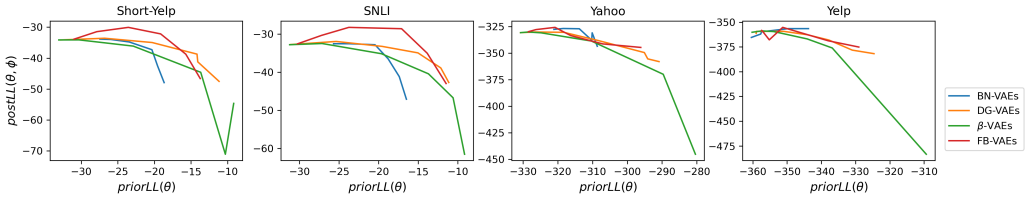

Figure 2: The curves of $priorLL(\boldsymbol{\theta})$ and $postLL(\boldsymbol{\theta}, \boldsymbol{\phi})$ in Gaussian distribution-based VAEs.

We also compare the performances of DG-vMF-VAEs with vMF-VAEs under different settings of κ. As they have the same $KL(\boldsymbol{\phi})$ while $AU(\boldsymbol{\phi})$ and $CU(\boldsymbol{\phi})$ are inappropriate to report for vMF distributions, we only plot their curves of $priorLL(\boldsymbol{\theta})$ and $postLL(\boldsymbol{\theta}, \boldsymbol{\phi})$ in Figure 3. It can be observed that DG-vMF-VAEs outperform vMF-VAEs in most cases.

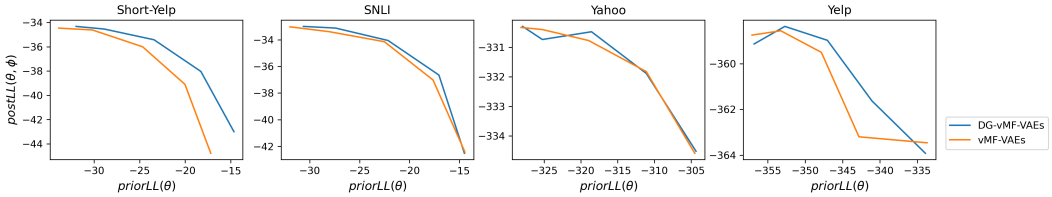

Figure 3: The curves of $priorLL(\boldsymbol{\theta})$ and $postLL(\boldsymbol{\theta}, \boldsymbol{\phi})$ in vMF distribution-based VAEs.

4.3 Visualization of the Posterior

To further investigate the posterior in latent space of those models, we visualize the aggregated posterior distribution and the posterior centers distribution on the 2 most active dimensions, i.e., the two dimensions with the highest $Var_{\mathbf{x}\sim\mathbf{X}}[\mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{z}|\mathbf{x})}[\mathbf{z}]]$, as depicted in Figure 4.

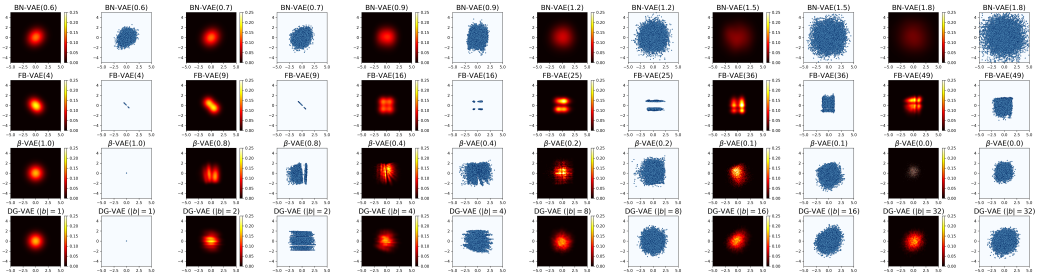

Figure 4: The visualization of aggregated posterior distribution (red-in-black) and posterior centers distribution (blue-in-white) for BN-VAEs, FB-VAEs, β-VAEs and DG-VAEs on Yahoo test-set. Illustrations for more datasets, more models and more dimensions are shown in Appendix G.

[3]There's a little difference between DG-VAE ($|b| = 32$) and DG-VAE (default): DG-VAE ($|b| = 32$) ignores data batch B if $|B| < 32$ while DG-VAE (default) accepts it through adapting to its batch size.

Here we can observe that BN-VAEs, FB-VAEs and β-VAEs can better solve the posterior collapse with harder restrictions or further weakening, but meanwhile they are faced with different kind of mismatch between the aggregated posterior distribution and the prior distribution, i.e., hole problem. In contrast, with the increase of aggregation size $|b|$, DG-VAE can better solve posterior collapse and avoid hole problem at the meantime.

4.4 Interpolation Study

One of the main advantages of VAEs over normal language models (e.g., GPT-2[24]) is that VAEs embed datapoints into a continuous latent space and enables latent-guided generation. We evaluate this ability through interpolation, where the models encode two sentences $\mathbf{x}_a$ and $\mathbf{x}_b$ as their posterior centers, i.e., $z_a = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_a)}[\mathbf{z}]$ and $z_b = \mathbb{E}_{q_\phi(\mathbf{z}|\mathbf{x}_b)}[\mathbf{z}]$, and decode the variables between them, i.e., $z_\lambda = z_a \cdot (1-\lambda) + z_b \cdot \lambda, \lambda \in \{0.0, 0.1, \ldots, 1.0\}$.[4] The interpolated sentences are wished to be semantically smooth and meaningful, which we evaluate through the average Rouge-L F1-score [22], as stated in Eq. 14, where F_{lcs} denotes the F1-score of Longest Common Subsequence (LCS). We plot the curves of Rouge-L F1-score and λ for models on Yahoo dataset in Figure 5 and those curves on other datasets in the Appendix E.

$$RougeL_{F1}(\mathbf{y}_a, \mathbf{y}_b, \mathbf{y}_\lambda) = \frac{1}{2}(F_{lcs}(\mathbf{y}_a, \mathbf{y}_\lambda) + F_{lcs}(\mathbf{y}_b, \mathbf{y}_\lambda)) \tag{14}$$

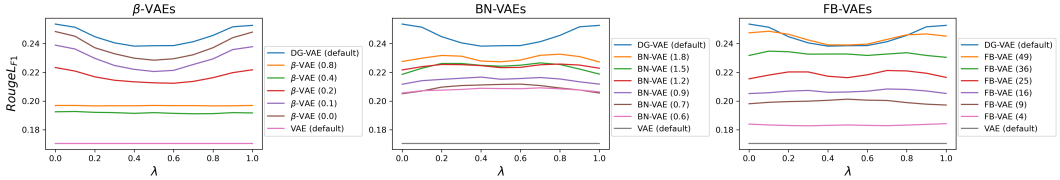

Figure 5: The curves of Rouge-L F1-score and λ for models' interpolation performance on Yahoo.

As shown in Figure 5, the average F1-score of LCS tend to be lower in the middle than at the ends, which indicates that generated sentences tend to be not smooth at the middle, which corresponds to the phenomenon of generation near to holes observed in previous work [21].[5] The vanilla VAE performs the worst as it suffers from posterior collapse, and only generates the same plain sentence; meanwhile, DG-VAE outperforms BN-VAEs, FB-VAEs and β-VAEs in the quality of interpolation on Yahoo dataset as it can solve posterior collapse and avoid hole problem at the same time.

In summary, the existing methods for solving posterior collapse in VAEs either have limited effect, or can effectively solve posterior collapse at the cost of introducing hole problem. In contrast, our proposed DG-VAE can effectively solve posterior collapse and avoid hole problem at the same time, which is demonstrated by the posterior centers spread in latent space and the aggregated posterior in consistent with the prior. Furthermore, our proposed DG-VAE outperforms the existing methods in the quality of latent-guided generation due to these improvements in latent space.

5 Discussion

Conclusion In this work, we take systematic experiments to demonstrate the posterior collapse and hole problem in existing continuous VAEs for text generation. To solve both problems at the same time, we propose a density gap-based regularization on the aggregated posterior to replace the KL regularization in ELBo, and prove it in essence maximizes the ELBo as well as the mutual information between the latent and the input. Experiments on real-world datasets prove the effectiveness of our method on solving both problems and its improvement on latent-guided generation.

Limitation & Future work Both the theory and the ablation study show that the effectiveness of our proposed method depends on the aggregation size $|b|$, which is still limited by the batch size during training. Therefore a promising future direction is to find a solution to break this limit, such like the memory bank mechanism in contrastive learning [31].

[4] We only consider greedy search for generation in this work.

[5] For further illustration on this phenomenon, we provide case study in Appendix F.

References

[1] Jyoti Aneja, Alexander G. Schwing, Jan Kautz, and Arash Vahdat. A contrastive learning approach for training variational autoencoder priors. In Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan, editors, *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, pages 480–493, 2021.

[2] Hareesh Bahuleyan, Lili Mou, Hao Zhou, and Olga Vechtomova. Stochastic wasserstein autoencoder for probabilistic sentence generation. In Jill Burstein, Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4068–4076. Association for Computational Linguistics, 2019.

[3] Matthias Bauer and Andriy Mnih. Resampled priors for variational autoencoders. In Kamalika Chaudhuri and Masashi Sugiyama, editors, *The 22nd International Conference on Artificial Intelligence and Statistics, AISTATS 2019, 16-18 April 2019, Naha, Okinawa, Japan*, volume 89 of *Proceedings of Machine Learning Research*, pages 66–75. PMLR, 2019.

[4] Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. A large annotated corpus for learning natural language inference. In Lluís Màrquez, Chris Callison-Burch, Jian Su, Daniele Pighin, and Yuval Marton, editors, *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 632–642. The Association for Computational Linguistics, 2015.

[5] Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew M. Dai, Rafal Józefowicz, and Samy Bengio. Generating sentences from a continuous space. In Yoav Goldberg and Stefan Riezler, editors, *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning, CoNLL 2016, Berlin, Germany, August 11-12, 2016*, pages 10–21. ACL, 2016.

[6] Yuri Burda, Roger B. Grosse, and Ruslan Salakhutdinov. Importance weighted autoencoders. In Yoshua Bengio and Yann LeCun, editors, *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016.

[7] Wenqing Chen, Jidong Tian, Liqiang Xiao, Hao He, and Yaohui Jin. A semantically consistent and syntactically variational encoder-decoder framework for paraphrase generation. In Donia Scott, Núria Bel, and Chengqing Zong, editors, *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 1186–1198. International Committee on Computational Linguistics, 2020.

[8] Tim R. Davidson, Luca Falorsi, Nicola De Cao, Thomas Kipf, and Jakub M. Tomczak. Hyperspherical variational auto-encoders. In Amir Globerson and Ricardo Silva, editors, *Proceedings of the Thirty-Fourth Conference on Uncertainty in Artificial Intelligence, UAI 2018, Monterey, California, USA, August 6-10, 2018*, pages 856–865. AUAI Press, 2018.

[9] Adji B. Dieng, Yoon Kim, Alexander M. Rush, and David M. Blei. Avoiding latent variable collapse with generative skip models. In Kamalika Chaudhuri and Masashi Sugiyama, editors, *The 22nd International Conference on Artificial Intelligence and Statistics, AISTATS 2019, 16-18 April 2019, Naha, Okinawa, Japan*, volume 89 of *Proceedings of Machine Learning Research*, pages 2397–2405. PMLR, 2019.

[10] Le Fang, Chunyuan Li, Jianfeng Gao, Wen Dong, and Changyou Chen. Implicit deep latent variable models for text generation. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3944–3954. Association for Computational Linguistics, 2019.

[11] Hao Fu, Chunyuan Li, Xiaodong Liu, Jianfeng Gao, Asli Celikyilmaz, and Lawrence Carin. Cyclical annealing schedule: A simple approach to mitigating KL vanishing. In Jill Burstein,

Christy Doran, and Thamar Solorio, editors, *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 240–250. Association for Computational Linguistics, 2019.

[12] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron C. Courville, and Yoshua Bengio. Generative adversarial nets. In Zoubin Ghahramani, Max Welling, Corinna Cortes, Neil D. Lawrence, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 2672–2680, 2014.

[13] Ishaan Gulrajani, Kundan Kumar, Faruk Ahmed, Adrien Ali Taïga, Francesco Visin, David Vázquez, and Aaron C. Courville. Pixelvae: A latent variable model for natural images. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[14] Irina Higgins, Loïc Matthey, Arka Pal, Christopher Burgess, Xavier Glorot, Matthew Botvinick, Shakir Mohamed, and Alexander Lerchner. beta-vae: Learning basic visual concepts with a constrained variational framework. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net, 2017.

[15] Matthew D Hoffman and Matthew J Johnson. Elbo surgery: yet another way to carve up the variational evidence lower bound. In *Workshop in Advances in Approximate Bayesian Inference, NIPS*, volume 1, 2016.

[16] Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P. Xing. Toward controlled generation of text. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 1587–1596. PMLR, 2017.

[17] Diederik P. Kingma, Tim Salimans, Rafal Józefowicz, Xi Chen, Ilya Sutskever, and Max Welling. Improving variational autoencoders with inverse autoregressive flow. In Daniel D. Lee, Masashi Sugiyama, Ulrike von Luxburg, Isabelle Guyon, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 29: Annual Conference on Neural Information Processing Systems 2016, December 5-10, 2016, Barcelona, Spain*, pages 4736–4744, 2016.

[18] Diederik P. Kingma and Max Welling. Auto-encoding variational bayes. In Yoshua Bengio and Yann LeCun, editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*, 2014.

[19] Alexej Klushyn, Nutan Chen, Richard Kurle, Botond Cseke, and Patrick van der Smagt. Learning hierarchical priors in vaes. In Hanna M. Wallach, Hugo Larochelle, Alina Beygelzimer, Florence d'Alché-Buc, Emily B. Fox, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 2866–2875, 2019.

[20] Bohan Li, Junxian He, Graham Neubig, Taylor Berg-Kirkpatrick, and Yiming Yang. A surprisingly effective fix for deep latent variable modeling of text. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 3601–3612. Association for Computational Linguistics, 2019.

[21] Ruizhe Li, Xutan Peng, and Chenghua Lin. On the latent holes of vaes for text generation. *CoRR*, abs/2110.03318, 2021.

[22] Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics.

[23] Alireza Makhzani, Jonathon Shlens, Navdeep Jaitly, and Ian J. Goodfellow. Adversarial autoencoders. *CoRR*, abs/1511.05644, 2015.

[24] Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[25] Ali Razavi, Aäron van den Oord, Ben Poole, and Oriol Vinyals. Preventing posterior collapse with delta-vaes. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net, 2019.

[26] Danilo Jimenez Rezende, Shakir Mohamed, and Daan Wierstra. Stochastic backpropagation and approximate inference in deep generative models. In *Proceedings of the 31th International Conference on Machine Learning, ICML 2014, Beijing, China, 21-26 June 2014*, volume 32 of *JMLR Workshop and Conference Proceedings*, pages 1278–1286. JMLR.org, 2014.

[27] Danilo Jimenez Rezende and Fabio Viola. Taming vaes. *CoRR*, abs/1810.00597, 2018.

[28] Harshil Shah and David Barber. Generative neural machine translation. In Samy Bengio, Hanna M. Wallach, Hugo Larochelle, Kristen Grauman, Nicolò Cesa-Bianchi, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 31: Annual Conference on Neural Information Processing Systems 2018, NeurIPS 2018, December 3-8, 2018, Montréal, Canada*, pages 1353–1362, 2018.

[29] Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi S. Jaakkola. Style transfer from non-parallel text by cross-alignment. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 6830–6841, 2017.

[30] Ilya O. Tolstikhin, Olivier Bousquet, Sylvain Gelly, and Bernhard Schölkopf. Wasserstein auto-encoders. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net, 2018.

[31] Zhirong Wu, Yuanjun Xiong, Stella X. Yu, and Dahua Lin. Unsupervised feature learning via non-parametric instance discrimination. In *2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018*, pages 3733–3742. Computer Vision Foundation / IEEE Computer Society, 2018.

[32] Haowen Xu, Wenxiao Chen, Jinlin Lai, Zhihan Li, Youjian Zhao, and Dan Pei. On the necessity and effectiveness of learning the prior of variational auto-encoder. *CoRR*, abs/1905.13452, 2019.

[33] Jiacheng Xu and Greg Durrett. Spherical latent spaces for stable variational autoencoders. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 4503–4513. Association for Computational Linguistics, 2018.

[34] Zichao Yang, Zhiting Hu, Ruslan Salakhutdinov, and Taylor Berg-Kirkpatrick. Improved variational autoencoders for text modeling using dilated convolutions. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 3881–3890. PMLR, 2017.

[35] Kang Min Yoo, Youhyun Shin, and Sang-goo Lee. Data augmentation for spoken language understanding via joint variational generation. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 7402–7409. AAAI Press, 2019.

[36] Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In Corinna Cortes, Neil D. Lawrence, Daniel D. Lee, Masashi Sugiyama, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 28: Annual Conference on Neural Information Processing Systems 2015, December 7-12, 2015, Montreal, Quebec, Canada*, pages 649–657, 2015.

[37] Tiancheng Zhao, Ran Zhao, and Maxine Eskénazi. Learning discourse-level diversity for neural dialog models using conditional variational autoencoders. In Regina Barzilay and Min-Yen Kan, editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 654–664. Association for Computational Linguistics, 2017.

[38] Qile Zhu, Wei Bi, Xiaojiang Liu, Xiyao Ma, Xiaolin Li, and Dapeng Wu. A batch normalized inference network keeps the KL vanishing away. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 2636–2649. Association for Computational Linguistics, 2020.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] We propose a novel regularization for VAEs to solve both posterior collapse and hole problem.
   (b) Did you describe the limitations of your work? [Yes] See Section 5.
   (c) Did you discuss any potential negative societal impacts of your work? [No]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] We state the full set of assumptions in the first paragraph of Section 3.
   (b) Did you include complete proofs of all theoretical results? [Yes] We provide complete proofs for all theoretical results in Section 3.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] We include the code, scripts for downloading data and instructions for reproduction in the supplemental material.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Appendix B.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [No]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Appendix B.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See Section 4.
   (b) Did you mention the license of the assets? [No]
   (c) Did you include any new assets either in the supplemental material or as a URL? [No]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [No]
5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]