Addressing Posterior Collapse with Mutual Information for Improved Variational Neural Machine Translation

Arya D. McCarthy[♠◇] and **Xian Li**[◇] and **Jiatao Gu**[◇] and **Ning Dong**[◇]
[♠]Johns Hopkins University
[◇]Facebook
arya@jhu.edu, {xianl,jgu,dnn}@fb.com

Abstract

This paper proposes a simple and effective approach to address the problem of posterior collapse in conditional variational autoencoders (CVAEs). It thus improves performance of machine translation models that use noisy or monolingual data, as well as in conventional settings. Extending Transformer and conditional VAEs, our proposed latent variable model measurably prevents posterior collapse by (1) using a modified evidence lower bound (ELBO) objective which promotes mutual information between the latent variable and the target, and (2) guiding the latent variable with an auxiliary bag-of-words prediction task. As a result, the proposed model yields improved translation quality compared to existing variational NMT models on WMT Ro↔En and De↔En. With latent variables being effectively utilized, our model demonstrates improved robustness over non-latent Transformer in handling uncertainty: exploiting noisy source-side monolingual data (up to +3.2 BLEU), and training with weakly aligned web-mined parallel data (up to +4.7 BLEU).

1 Introduction

The **conditional variational autoencoder** (CVAE; Sohn et al., 2015) is a conditional generative model for structured prediction tasks like machine translation. This model, learned by variational Bayesian methods (Kingma and Welling, 2014), can capture global signal about the target in its latent variables. Unfortunately, variational inference for text generation often yields models that ignore their latent variables (Bowman et al., 2016), a phenomenon called **posterior collapse**.

In this paper, we introduce a new loss function for CVAEs that counteracts posterior collapse, motivated by our analysis of CVAE's evidence lower bound objective (ELBO). Our analysis (§2) reveals that optimizing ELBO's second term not only brings the variational posterior approximation closer to the prior, but also decreases mutual information between latent variables and observed data. Based on this insight, we modify CVAE's ELBO in two ways (§3): (1) We explicitly add a principled mutual information term back into the training objective, and (2) we use a factorized decoder (Chen et al., 2017), which also predicts the target bag-of-words as an auxiliary decoding distribution to regularize our latent variables. Our objective is effective even without *Kullback–Leibler term (KL) annealing* (Bowman et al., 2016), a strategy for iteratively altering ELBO over the course of training to avoid posterior collapse.

In applying our method to neural machine translation (NMT; Kalchbrenner and Blunsom, 2013; Sutskever et al., 2014), we find that we have measurably mitigated posterior collapse. The latent variables are not ignored, even in the presence of a powerful Transformer decoder. By addressing this problem, the resulting NMT model has improved robustness and performance in low-resource scenarios. Noisy data like those scraped from the Internet (Smith et al., 2013; Michel and Neubig, 2018) present a challenge for NMT (Khayrallah and Koehn, 2018; Ott et al., 2018a); we are measurably more able to model this extrinsic uncertainty than the (non-latent) Transformer (Vaswani et al., 2017) or existing variational NMT with the CVAE architecture (Zhang et al., 2016). Finally, we extend the model to semi-supervised learning (Cheng et al., 2016) to more effectively learn from monolingual data.

In summary, our conditional text generation model overcomes posterior collapse by promoting mutual information. It can easily and successfully integrate noisy and monolingual data, and it does this without the cost of lower BLEU score than non-latent NMT in typical settings.

2 Formalism and Mathematical Analysis

Here we review the standard framework for neural MT. Next, we connect this to the conditional variational autoencoder, a model with latent random variables whose distributions are learned by black-box variational Bayesian inference. Finally, we analyze the CVAE's objective to explain why these models will ignore their latent variables ("posterior collapse").

2.1 Neural Machine Translation

Problem instances in machine translation are pairs of sequences ($\boldsymbol{x} \triangleq [x_1, \ldots, x_m], \boldsymbol{y} \triangleq [y_1, \ldots, y_n]$), where $\boldsymbol{x}$ and $\boldsymbol{y}$ represent the source and target sentences, respectively. Conventionally, a neural machine translation model is a parameterized conditional distribution whose likelihood factors in an autoregressive fashion:

$$p_{\boldsymbol{\theta}}(\boldsymbol{y} \mid \boldsymbol{x}) = \prod_{t=1}^{n} p_{\boldsymbol{\theta}}(y_t \mid \boldsymbol{x}, \boldsymbol{y}_{<t}) \,. \tag{1}$$

The dominant translation paradigm first represents the source sentence as a sequence of contextualized vectors (using the *encoder*), then decodes this representation into a target hypothesis according to Equation 1. The parameters $\boldsymbol{\theta}$ are learned by optimizing the log-likelihood of training pairs with stochastic gradient methods (Bottou and Cun, 2004; Kingma and Ba, 2015). Decoding is deterministic, using an efficient approximate search like beam search (Tillmann and Ney, 2003). The Transformer architecture with multi-head attention has become the state of the art for NMT (Vaswani et al., 2017).

2.2 The Conditional Variational Autoencoder

Our NMT approach extends the conditional variational autoencoder (Sohn et al., 2015), which we identify as a generalization of Variational NMT (Zhang et al., 2016). It introduces a latent random variable $\boldsymbol{z}$ into the standard NMT conditional distribution from Equation 1:[1,2]

$$p_{\boldsymbol{\theta}}(\boldsymbol{y} \mid \boldsymbol{x}) = \int_{\boldsymbol{z}} \underbrace{p_{\boldsymbol{\theta}}(\boldsymbol{y} \mid \boldsymbol{z}, \boldsymbol{x})}_{\text{decoder}} \cdot \underbrace{p_{\boldsymbol{\theta}}(\boldsymbol{z} \mid \boldsymbol{x})}_{\text{encoder}} \, \mathrm{d}\boldsymbol{z}. \tag{2}$$

For a given source sentence $\boldsymbol{x}$, first a latent variable $\boldsymbol{z}$ is sampled from the encoder, then the target sentence $\boldsymbol{y}$ is generated by the decoder: $\boldsymbol{z} \sim p_{\boldsymbol{\theta}}(\boldsymbol{z} \mid \boldsymbol{x}), \boldsymbol{y} \sim p_{\boldsymbol{\theta}}(\boldsymbol{y} \mid \boldsymbol{z}, \boldsymbol{x})$.[3]

It is intractable to marginalize Equation 2 over $\boldsymbol{z}$. Instead, the CVAE training objective is a variational lower bound (the ELBO) of the conditional log-likelihood. It relies on a parametric approximation of the model posterior: $q_{\boldsymbol{\phi}}(\boldsymbol{z} \mid \boldsymbol{x}, \boldsymbol{y})$. The variational family we choose for q is a neural network whose parameters $\boldsymbol{\phi}$ are shared (i.e., amortized) across the dataset.

The ELBO lower-bounds the log-likelihood, as can be proven with Jensen's inequality. Its form is:

$$\begin{aligned}\mathcal{L}_{\text{CVAE}} = \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{y})} \left[\log p_{\boldsymbol{\theta}}(\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{z})\right] \\ - \mathrm{D}_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\boldsymbol{z} \mid \boldsymbol{x}, \boldsymbol{y}) \parallel p_{\boldsymbol{\theta}}(\boldsymbol{z} \mid \boldsymbol{x})),\end{aligned} \tag{3}$$

where D_{KL} represents the Kullback–Leibler divergence between two distributions.

We use amortized variational inference to simultaneously perform learning and approximate posterior inference, updating both $\boldsymbol{\theta}$ and $\boldsymbol{\phi}$ with stochastic gradient methods. Improving $\boldsymbol{\theta}$ raises the lower bound, and improving $\boldsymbol{\phi}$ keeps the bound tight with respect to the model conditional log-likelihood. The same argument pertains to the joint maximization interpretation of the expectation–maximization (EM) algorithm (Neal and Hinton, 1998). (Our optimization is a variational generalization of EM.)

2.3 Posterior Collapse

Despite their success when applied to computer vision tasks, variational autoencoders in natural language generation suffer from **posterior collapse**, where the learnt latent code is ignored by a strong autoregressive decoder. This presents a challenge to conditional language generation tasks in NLP like machine translation.

The phenomenon can be explained mathematically by an analysis of the ELBO objective, as well as from the perspective of a powerful decoder that can model the true distribution without needing the latent code. We consider both in this subsection.

ELBO surgery Recall that the computed objective approximates the objective on the true data distribution $p_{\mathcal{D}}$, using a finite number of samples

[1] By contrast, the hidden states of a standard sequence-to-sequence model are *deterministic* latent variables.

[2] In Equation 2 we assume a continuous latent variable. For the discrete case, replace integration with summation.

[3] The sense of "encoder" in the context of variational autoencoders differs from the typical sense in neural machine translation, such that the NMT encoder is a component of both the VAE's encoder and decoder. We can separate these by computing a second, deterministic latent variable $\boldsymbol{h}$ from $\boldsymbol{x}$ to represent the NMT encoder outputs, used by both the VAE encoder and the NMT/VAE decoder.

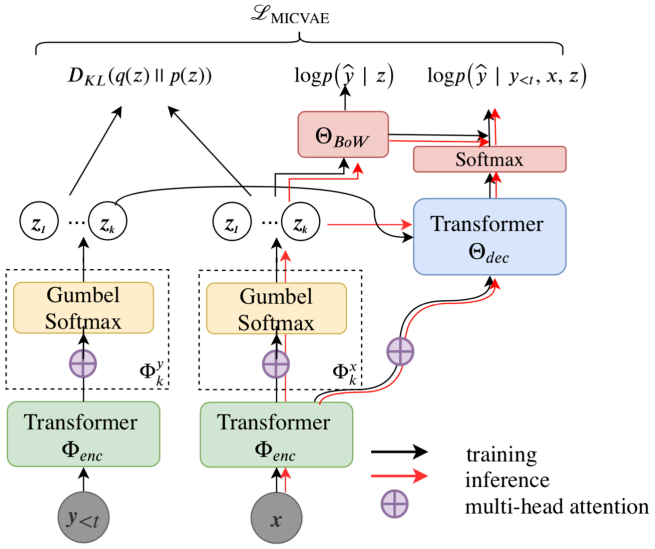

Figure 1: Model architecture in training (with parallel data) and inference.

(see, e.g., Brown et al., 1992):

$$\mathcal{L} = \mathbb{E}_{p_{\mathcal{D}}(\boldsymbol{x},\boldsymbol{y})} \left[\mathcal{L}_{\text{CVAE}}(\boldsymbol{\phi}, \boldsymbol{\theta}; \boldsymbol{x}, \boldsymbol{y})\right]. \quad (4)$$

We can factor the KL term of Equation 3 (omitting parameter subscripts) as:

$$\begin{aligned} &\mathbb{E}_{p_{\mathcal{D}}(\boldsymbol{x},\boldsymbol{y})} \left[\mathrm{D}_{\mathrm{KL}}(q(\boldsymbol{z} \mid \boldsymbol{x}, \boldsymbol{y}) \parallel p(\boldsymbol{z} \mid \boldsymbol{x}))\right] \\ &= \underbrace{\mathrm{H}(\boldsymbol{x}, \boldsymbol{y}) - \mathrm{H}(\boldsymbol{x}, \boldsymbol{y} \mid \boldsymbol{z})}_{\triangleq \mathrm{I}_{q_\phi}(\boldsymbol{z};\boldsymbol{x},\boldsymbol{y})} + \underbrace{\mathbb{E}_{q(\boldsymbol{z})} \log \frac{q(\boldsymbol{z})}{p(\boldsymbol{z})}}_{\triangleq \mathrm{D}_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\boldsymbol{z}) \| p(\boldsymbol{z}))}, \end{aligned} \quad (5)$$

which we prove in Appendix A, following (Hoffman and Johnson, 2016).

As both the resulting mutual information and KL terms are non-negative (Cover and Thomas, 2006), the global minimum of Equation 5 is $\mathrm{I}_{q_\phi}(\boldsymbol{z}; \boldsymbol{x}, \boldsymbol{y}) = \mathrm{D}_{\mathrm{KL}}(q_\phi(\boldsymbol{z}) \parallel p(\boldsymbol{z})) = 0$. Unfortunately, at this point, the consequence of the optimization is that the latent variable $\boldsymbol{z}$ is conditionally independent of the data $(\boldsymbol{x}, \boldsymbol{y})$.

A powerful decoder Revisiting Equation 3, we see that the decoder is conditioned on both the stochastic latent variable $\boldsymbol{z}$ and the source text $\boldsymbol{x}$. A sufficiently high-capacity autoregressive decoder can model the conditional density directly, ignoring the latent variable and reducing inference to Equation 1. The KL term can then be reduced to its minimum (0) by equating the posterior to the prior. To prevent this, some work weakens the decoder in various ways. This is a challenge, because NMT requires a powerful decoder such as Transformer with direct attention to the encoder.

3 An Information-Infused Objective

We modify our training objective to explicitly retain mutual information between the latent variable $\boldsymbol{z}$ and the observation $(\boldsymbol{x}, \boldsymbol{y})$. Further, we use an auxiliary decoder that only uses the latent variable, not the encoder states. We combine it with the existing decoder as a mixture of softmaxes (Yang et al., 2018a). The model is trained with amortized variational inference. When source-language monolingual text is available, we augment our modified CVAE objective with a similarly modified (non-conditional) VAE objective. The training and inference strategy is summarized in Figure 1.

3.1 Adding $\mathrm{I}_{q_\phi}(\boldsymbol{z}; \boldsymbol{x}, \boldsymbol{y})$ to ELBO

To combat the optimization dilemma from Equation 5 (namely, that the objective discourages mutual information between the latent variable and the data), we explicitly add the mutual information term to the CVAE's ELBO and obtain a new training objective:

$$\begin{aligned} \mathcal{L}_{\text{MICVAE}} &= \mathcal{L}_{\text{CVAE}} + \mathrm{I}_{q_\phi}(\boldsymbol{z}; \boldsymbol{x}, \boldsymbol{y}) \\ &= \mathbb{E}_{q_{\boldsymbol{\phi}}(\boldsymbol{z}|\boldsymbol{x},\boldsymbol{y})} \log p(\boldsymbol{y} \mid \boldsymbol{x}, \boldsymbol{z}) \\ &\quad - \mathrm{D}_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\boldsymbol{z}) \parallel p(\boldsymbol{z})) \end{aligned} \quad (6)$$

The new training objective $\mathcal{L}_{\text{MICVAE}}$ aims to match the aggregated approximate posterior distribution of the latent variable $q_\phi(\boldsymbol{z})$ (Hoffman and Johnson, 2016) to the aggregated-posterior prior distribution $p_{\boldsymbol{\theta}}(\boldsymbol{z})$.[4]

[4]It can be seen as extending InfoVAE (Zhao et al., 2019) to conditional generative models, where we have overcome

3.2 Guiding z to Encode Global Information

Several existing approaches *weaken the decoder*: limiting its capacity to encourage latent variables to be utilized (Bowman et al., 2016; Gulrajani et al., 2017). Here we propose a different approach: explicitly guiding the information encoded in $\boldsymbol{z}$ without reducing the decoder's capacity.

The decision to weaken the decoder can be understood in the context of Bits-Back Coding theory (Chen et al., 2017), which suggests that at optimality the decoder will model whatever it can locally, and only the residual will be encoded in the latent variable $\boldsymbol{z}$. A consequence is that explicit information placement can give more powerful latent representations.

Inspired by this Bits-Back perspective, we add a global auxiliary loss for $\boldsymbol{z}$ to encode information which cannot be modelled locally by the autoregressive decoder $\prod_t p_{\boldsymbol{\theta}}(y_t \mid \boldsymbol{x}, \boldsymbol{y}_{<t}, \boldsymbol{z})$. We use bag-of-words (BoW) prediction as the auxiliary loss. It encodes global information while having a non-autoregressive factorization: $\prod_t p_{\boldsymbol{\psi}}(y_t \mid \boldsymbol{z})$. (We choose not to condition it on the source sentence $\boldsymbol{x}$.) Further, it requires no additional annotated data. The auxiliary decoder complements the autoregressive decoder (which is locally factorized), interpolating predictions at the softmax layer, i.e. $p(y_t \mid \boldsymbol{x}, \boldsymbol{y}_{<t}, \boldsymbol{z})$ is a **mixture of softmaxes** (Yang et al., 2018b):

$$p(y_t \mid \cdot) = (1-\lambda) \cdot p_{\boldsymbol{\theta}}(y_t \mid \boldsymbol{x}, \boldsymbol{y}_{<t}, \boldsymbol{z}) + \lambda \cdot p_{\boldsymbol{\psi}}(y_t \mid \boldsymbol{z}), \quad (7)$$

with mixing parameter λ. (We use $\lambda = 0.1$ in this paper.) Thus, the bag-of-words objective regularizes the log-likelihood bound.

4 Implementing Latent Variable NMT

4.1 Architecture

Our model uses discrete latent variables. These are used to select a latent embedding, which is concatenated to the decoder state.

Inference Network We use discrete latent variables with reparameterization via Gumbel-Softmax (Jang et al., 2017; Maddison et al., 2017) to allow backpropagation through discrete sampling. Unlike the multivariate Gaussian distribution commonly used in VAE and CVAE, our parameterization can explicitly account for multiple modes in the data. (See Rezende and Mohamed (2015) for a perspective on the value of multimodal distributions over latent variables.) To make our model more general, we introduce a *set* of discrete latent variables $\boldsymbol{z} = \{\boldsymbol{z}_1, \ldots, \boldsymbol{z}_K\}$ which are independently sampled from their own inference networks Φ_k. Specifically, each Φ_k computes scaled dot product attention with encoder outputs $\boldsymbol{h} \in \mathbb{R}^d$ using latent code embedding $\boldsymbol{e}_k$:

$$\begin{aligned} \boldsymbol{C}_k &= \text{Attention}\left(\boldsymbol{e}_k \boldsymbol{W}^k, \boldsymbol{h}\boldsymbol{W}^h, \boldsymbol{h}\boldsymbol{W}^h\right) \\ &= \text{Softmax}\left(\frac{\boldsymbol{e}_k \boldsymbol{W}^k (\boldsymbol{h}\boldsymbol{W}^h)^\top}{\sqrt{d}}\right) \boldsymbol{h}\boldsymbol{W}^h. \end{aligned} \quad (8)$$

We can now sample $\boldsymbol{z}_k$ by the Gumbel-Softmax reparameterization trick (Maddison et al., 2017; Jang et al., 2017):

$$\boldsymbol{z}_k \sim \text{GumbelSoftmax}(\mathbf{C}_k) \quad (9)$$

$$= \text{Softmax}\left(\frac{\mathbf{C}_k + \boldsymbol{g}}{\tau}\right), \quad (10)$$

where $\boldsymbol{g} = -\log(-\log(\boldsymbol{u})), \boldsymbol{u} \sim \text{Uniform}$ is the Gumbel noise and τ is a fixed temperature. (We use $\tau = 1$ in this paper.) At inference time, we use a discrete version by directly sampling from the latent variable distribution.

BoW Auxiliary Decoder Given an inferred sample $\boldsymbol{z} \sim \Phi_k(\boldsymbol{h})$, the BoW decoder predicts all tokens at once without considering their order. We compute the cross-entropy loss for the predicted tokens over the output vocabulary space V:

$$\mathcal{L}_{\text{BoW}} = \sum_{i=1}^{|V|} p_i \log \hat{p}_\psi(y_i \mid \boldsymbol{z}), \quad \sum_{i=1}^{|V|} p_i = 1. \quad (11)$$

We take the (unnormalized) empirical distribution $\tilde{p}_i$ to be a token's frequency within a sentence normalized by its total frequency within a mini-batch, mitigating the effect of frequent (stop) words. This is then normalized over the sentence to sum to 1, giving values p_i. The model distribution $\hat{p}_\psi$ is computed by conditioning on the latent code only, without direct attention to encoder outputs. We use scaled dot-product attention between the latent embeddings and the target embeddings (each of dimensionality d, represented as a matrix E_V):

$$p_\psi(y_i \mid \boldsymbol{z}) = \text{Softmax}\left(\frac{\mathbf{e}(\boldsymbol{z}) E_V^\intercal}{\sqrt{d}}\right)_i. \quad (12)$$

the mismatch between the (joint) data distribution $p_{\mathcal{D}}(\boldsymbol{x}, \boldsymbol{y})$ and the (conditional) likelihood objective $p_{\boldsymbol{\theta}}(\boldsymbol{y} \mid \boldsymbol{x})$.

Algorithm 1 Training Strategy

$\Phi_{enc}, \Phi_{k=1,\ldots,K}, \Theta_{dec}, \Theta_{BoW} \leftarrow$ init.
while $\Theta_{enc}, \Theta_{dec}, \Theta_{BoW}, \Phi_{k=1,\ldots,K}$ have not converged **do**
  Sample $(\mathbf{x}, \mathbf{y})$ from D^{bitext}
  Compute $\mathcal{L}_{\text{MICVAE}}$ with Equation 6
  Train $\Phi_{enc}, \Theta_{dec}, \Phi_{k=1,\ldots,K}$ with $\mathcal{L}_{\text{MICVAE}}$
  Compute $\mathcal{L}_{\text{BoW}}$ with Equation 12
  Train $\Phi_{enc}, \Theta_{BoW}, \Phi_{k=1,\ldots,K}$ with $\mathcal{L}_{\text{BoW}}$
  if self_training **then**
    Sample $\mathbf{x}$ from D^{mono}
    Compute $\mathcal{L}_{\text{Mono}}$ with Equation 13
    Train $\Phi_{enc}, \Phi_{k=1,\ldots,K}$ with $\mathcal{L}_{\text{Mono}}$
  end if
end while

4.2 Training

For training with parallel data, we optimize $\mathcal{L}_{\text{MICVAE}}$. We draw samples $\boldsymbol{z}$ from the approximate posterior $q_{\boldsymbol{\phi}}(\boldsymbol{z} \mid \boldsymbol{x}, \boldsymbol{y})$ parameterized by the inference network, then feed the samples to both the autoregressive and auxiliary (BoW) decoders to get a Monte Carlo estimate of the gradient.

Estimating aggregated distributions We estimate $p_{\boldsymbol{\theta}}(\boldsymbol{z})$ and $q_{\boldsymbol{\phi}}(\boldsymbol{z})$ over each minibatch, following Zhao et al. (2018).

Semi-supervised learning We apply the same modification to VAE's ELBO, following Zhao et al. (2019). For jointly training with source-side monolingual data, we add $\mathrm{I}_{q_{\boldsymbol{\phi}}}(\boldsymbol{z}; \boldsymbol{x})$ to the ELBO, and for target-side monolingual data, we add $\mathrm{I}_{q_{\boldsymbol{\phi}}}(\boldsymbol{z}; \boldsymbol{y})$.[5] The joint objective sums the modified CVAE and VAE objectives:

$$\mathcal{L}_{\text{Mono}} = \log p(\boldsymbol{x} \mid \boldsymbol{z}) + D_{\text{KL}}\left(\frac{1}{L}\sum_{\ell=1}^{L} q_{\boldsymbol{\phi}}\left(\boldsymbol{z}^{(\ell)} \,\middle|\, \boldsymbol{x}^{(\ell)}\right) \,\middle\|\, \frac{1}{L}\sum_{\ell=1}^{L} p\left(\boldsymbol{z}^{(\ell)}\right)\right) \tag{13}$$

$$\mathcal{L}_{\text{Joint}} = \mathcal{L}_{\text{MICVAE}} + \mathcal{L}_{\text{Mono}}, \tag{14}$$

where L is the number of monolingual examples. Algorithm 1 describes the overall training strategy.

[5] Learning to copy the target text has proven useful for low-resource NMT (Currey et al., 2017).

5 Experiments and Results

Here we present empirical results on the Transformer architecture. We evaluate our model on four standard datasets and compare against three baselines. We use four measures to quantify posterior collapse, then examine translation quality (BLEU score) in standard fully supervised settings, a semi-supervised setting, and a fully supervised setting with noisy source text. Hyperparameters, regularization choices, and subword vocabulary information can be found in §5.3.

The results show that we have effectively addressed posterior collapse: latent variables are no longer ignored despite the presence of a powerful decoder. As a result, we outperform both the standard Transformer and the Transformer-based variational NMT approach, when using noisy data or source-language monolingual data.

5.1 Datasets

First, we evaluate our models on a standard high-resource and low-resource benchmark dataset from WMT. Second, we focus on situations where noisy or monolingual data is available. We note that low-resource scenarios and noisy data are two representative challenges in MT (Lopez and Post, 2013).

WMT14 German–English We use data from the WMT14 news translation shared task, which has 3.9M sentence pairs for training with the same BPE tokenization as in Gu et al. (2018).

WMT16 Romanian–English We use data from the WMT16 news translation shared task. We use the same BPE-preprocessed (Sennrich et al., 2016b) train, dev and test splits as in Gu et al. (2018) with 608k sentence pairs for training.

FLoRes Sinhala–English For this low-resource benchmark, we use the same preprocessed data as in Guzmán et al. (2019). There are 646k sentence pairs.

MT for Noisy Text (MTNT) French–English This dataset pairs web-scraped text from Reddit with professional translations. We use 30k subword units built jointly from source and target sentences and only keep sentences with less than 100 tokens. For training, there are 34,380 sentence pairs for English–French and 17,616 sentence pairs for French–English (Michel and Neubig, 2018). We also used

18,676 *monolingual* sentences per language from the same data source (Reddit).

5.2 Baselines

We compare our model to three baselines:

Non-latent This is a standard Transformer model without latent variables.

VNMT A CVAE model with Gaussian distribution as proposed in Variational NMT by Zhang et al. (2016), which we reimplement using Transformer. (Zhang et al. (2016) use a GRU-based recurrent model.)

DCVAE A CVAE model with the same discrete latent variable parameterization as ours but without the new objective (i.e., the mutual information term and bag-of-words regularizer).

5.3 Implementation details

All of our models build on Transformer. For WMT14 De–En and WMT16 Ro–En, we use the base configuration (Vaswani et al., 2017): 6 blocks, with 512-dimensional embedding, 2048-dimensional feed-forward network, and 8 attention heads. For FLoRes (low-resource) and MTNT (low-resource and noisy), we use a smaller Transformer: 4 layers, 256-dimensional embedding, 1024-dimensional inner layers, and 4 attention heads. Input and output embeddings are shared between the inference network and decoder. We use $T = 4$ categorical latent variables of dimension 16 (found by grid search on the dev set). Auxiliary bag-of-words predictions are combined with the decoder prediction with $\lambda = 0.1$. We optimize using Adam (Kingma and Ba, 2015) with $\beta_1 = 0.9$, $\beta_2 = 0.98$, $\epsilon = 1\text{E-}8$, weight decay of 0.001, and the warmup and learning rate schedule of Ott et al. (2018b). All models are trained on 8 NVIDIA V100 GPUs with 32K tokens per mini-batch. We train WMT14 De–En with 200k updates and others with 100k updates. We do not use early stopping.

We employ joint BPE vocabularies. The sizes are 32k for En–De and En–Ro; 30k for Fr–En; and 3k for Si–En. We also use a word dropout rate of 0.4 during training of all models, which is complementary to our approach.

We found the default initialization in the FAIRSEQ NMT toolkit was effective; we did not need to explore several initializations to avoid degenerate models.

Model	D_{KL}	$\mathrm{I}_{q_\phi}(\boldsymbol{z}, \boldsymbol{x})$	$\mathrm{I}_{q_\phi}(\boldsymbol{z}, \boldsymbol{y})$	NLL
DCVAE + KLA	0.001	0.001	4.2E-6	3.17
Our model	0.17	0.18	0.31	3.16

Table 1: Our model mitigates posterior collapse. The KL value refers to $\mathrm{D}_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\boldsymbol{z} \mid \boldsymbol{x}, \boldsymbol{y}) \parallel p_{\boldsymbol{\theta}}(\boldsymbol{z} \mid \boldsymbol{x}))$ for DCVAE and $\mathrm{D}_{\mathrm{KL}}(q_{\boldsymbol{\phi}}(\boldsymbol{z} \mid \boldsymbol{y}) \parallel p_{\boldsymbol{\theta}}(\boldsymbol{z} \mid \boldsymbol{x}))$ for our model.

5.4 Preventing Posterior Collapse

We compare our model to a standard DCVAE lacking the new objective. We report four metrics of posterior collapse on the validation set of WMT Ro–En:

1. Kullback–Leibler divergence (KL).
2. Mutual information between the latent variable and the source: $\mathrm{I}_{q_{\boldsymbol{\phi}}}(\boldsymbol{z}; \boldsymbol{x})$
3. Mutual information between the latent variable and the target: $\mathrm{I}_{q_{\boldsymbol{\phi}}}(\boldsymbol{z}; \boldsymbol{y})$.
4. Negative conditional log-likelihood (NLL) per token.

Table 1 shows that when using standard DCVAE ELBO, even with the common practice of KL annealing (KLA), both the KL loss and mutual information settle to almost 0 which is consistent with the analysis in Equation 5.

We also plot the progression of D_{KL}, $\mathrm{I}_{q_\phi}(\boldsymbol{z}; \boldsymbol{x})$, and $\mathrm{I}_{q_\phi}(\boldsymbol{z}; \boldsymbol{y})$ during training in Figure 2. The posterior collapse of the baseline model is apparent: both D_{KL} mutual information terms drop to 0 at the beginning of training as a result ELBO's design. On the other hand, our model, without using any annealing schedule, effectively increases mutual information and prevents KL loss from settling to a degenerate solution early on.

5.5 Translation Quality

We report corpus-level BLEU (Papineni et al., 2002)[6] on the test sets where the translations are generated by sampling each $\boldsymbol{z}_k$ with soft-assignment (vs. argmax).

Supervised Learning on Parallel Data First, we evaluate our model's performance when trained with parallel data on standard WMT datasets. Table 2 shows that our model consistently outperforms both VNMT and DCVAE models—which

[6]We use detokenized SacreBLEU (Post, 2018).

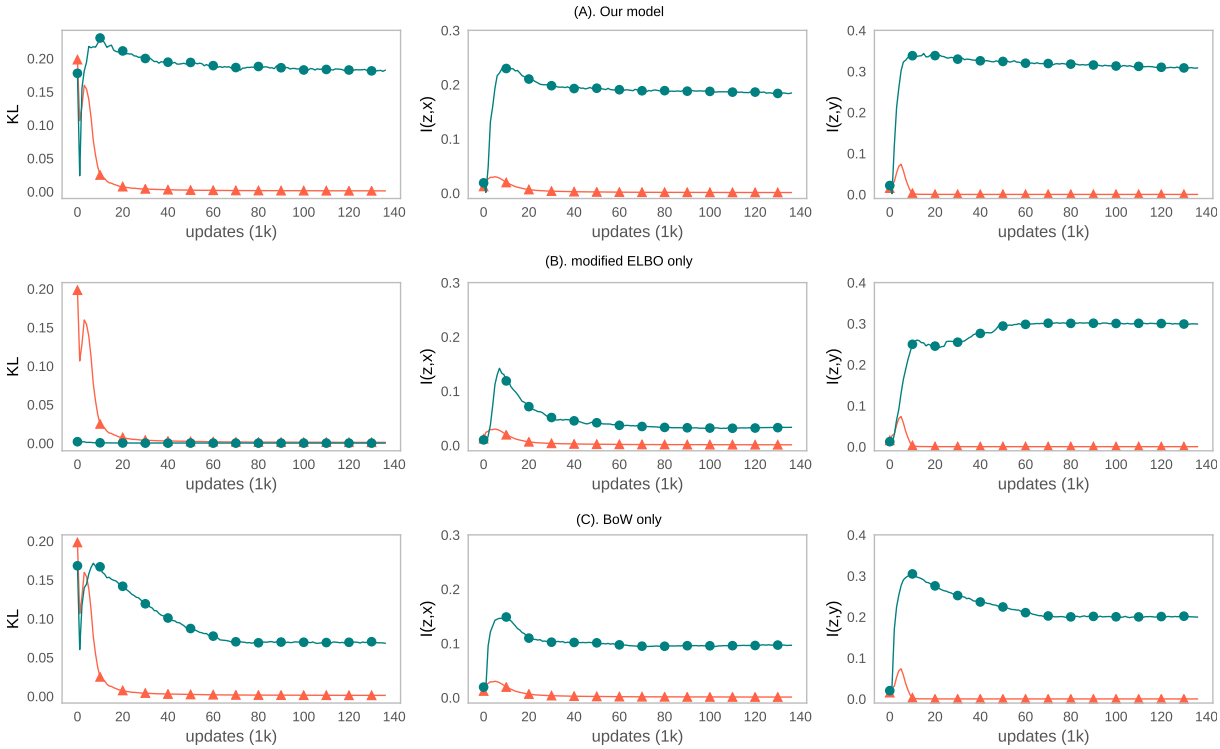

Figure 2: Row (A): comparison of KL and mutual information between baseline (DCVAE, solid triangle, orange color) and our model (solid circle, teal color). Rows (B) and (C): ablation study on relative contribution from MICVAE and BoW. All metrics are computed on the WMT16 Ro–En validation set over the course of 140k training updates.

	WMT16		WMT14	
Model	Ro–En	En–Ro	De–En	En–De
VNMT	34.20	34.27	30.35	25.84
DCVAE	34.16	34.51	29.76	25.46
Our model	**34.76**	**34.97**	**31.39**	**26.42**
Non-latent	34.73	34.54	30.89	26.36

Table 2: BLEU score on WMT benchmarks. Best result on each dataset is in bold. Our model provides minor gains (≤ 0.5 points) over the standard Transformer, not degrading like VNMT and DCVAE. Alongside improvements in semi-supervised or noisy settings, this suggests that there is no BLEU compromise in choosing this model.

Model	Fr–En	En–Fr
Non-latent	26.7	24.8
DCVAE	26.4	26.1
+ source mono	27.3	26.4
Our model	**28.6**	**26.3**
+ source mono	**29.8**	**26.7**

Table 3: Translation performance (BLEU) of utilizing source-side monolingual data. Best result on each data condition (with and without monolingual data) is bold.

require ad-hoc KL annealing—while on par with a strong Transformer baseline.

Semi-supervised with Source-side Monolingual Data Leveraging monolingual data is a common practice to improve low resource NMT. One popular approach uses target-side monolingual data through "backtranslation" as a data augmentation, but how to effectively leverage source-side monolingual data is an open challenge (Sennrich et al., 2016a; Zhang and Zong, 2016; Wu et al., 2019). We use the joint training objective described in Equation 14. To have a fair comparison, we also extend VNMT and DCVAE with the same joint training algorithm, i.e., the newly added monolingual data is used to train their corresponding sequence encoder and inference network with standard VAE ELBO. That is, the only difference is that our model was trained to promote mutual information $\mathrm{I}_{q_\phi}(\boldsymbol{z}, \boldsymbol{x})$ and $\mathrm{I}_{q_\phi}(\boldsymbol{z}, \boldsymbol{y})$. As shown in Table 3, by doing so the proposed model brings larger gains during semi-supervised learning with source-side monolingual data.

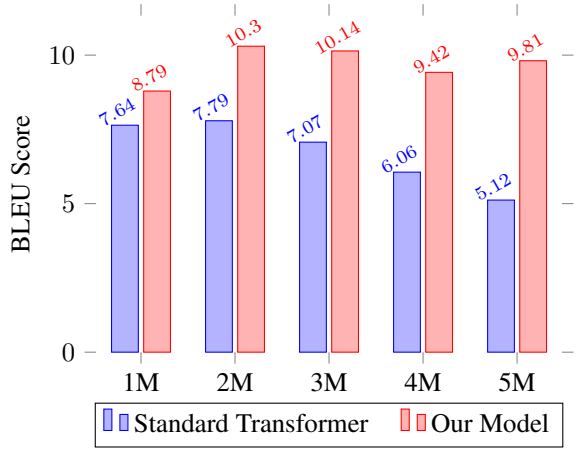

Figure 3: BLEU when increasing the number of noisy parallel sentences (ranked by Zipporah) in training, Si–En.

Robustness to Noisy Data While high-quality parallel data is scarce for low-resource language pairs, weakly aligned sentence pairs can be mined from massive unpaired data such as Paracrawl.[7] We evaluate our model's performance when augmenting the training set with increasingly noisy parallel data filtered by Zipporah (Xu and Koehn, 2017). Because VNMT and DCVAE underperform our proposal in previous experiments, we omit them from this experiment. Figure 3 shows the results in the Sinhala–English direction. Our model always outperforms standard Transformer, which struggles as more (and noisier) data is added. The gap grows from +1.2 to +4.7 BLEU.

6 Analysis

Ablation Study How do the different ingredients of our proposed approach contribute to preventing posterior collapse and improving translation quality? We explore two variants of the proposed model: 1) modified ELBO only: only adding mutual information term to the training objective, while without gradients from $\mathcal{L}_{\text{BoW}}$, 2) BoW only: which is equivalent to DCVAE combined with BoW decoder.

First, we perform the same collapse metrics evaluation as in Table 1. Figure 2(B) suggests that by explicitly adding mutual information term back to the training objective, both $\mathrm{I}_{q_\phi}(\boldsymbol{z};\boldsymbol{x})$ and $\mathrm{I}_{q_\phi}(\boldsymbol{z};\boldsymbol{y})$ are effectively raised, while the remaining aggregated KL term is still optimized to zero. Such behavior is consistent with the analysis revealed in Equation 5. On the other hand, regularizing $\boldsymbol{z}$ with the BoW decoder only, shown in Figure 2(C), is very effective in preventing KL vanishing as well as increasing mutual information. When two approaches are combined, as was shown in Figure 2(A), the model retains higher mutual information for both $\mathrm{I}_{q_\phi}(\boldsymbol{z};\boldsymbol{x})$ and $\mathrm{I}_{q_\phi}(\boldsymbol{z};\boldsymbol{y})$.

[7] https://paracrawl.eu/

Model	De–En (3.9M)	Ro–En (608K)
BoW and $\mathcal{L}_{\text{MICVAE}}$	**31.4**	**34.8**
BoW only	31.1	34.2

Table 4: Ablation study on translation quality (BLEU). The information-infused loss function provides additional performance over the DCVAE with a bag-of-words decoder.

Next, we see whether the difference in mutual information yields different translation quality. We compare two models: BoW only (Figure 2(C)) and both (Figure 2(A)), on WMT14 De–En and WMT16 Ro–En test sets. Table 4 shows the difference matters more in a low-data regime.

Analysis of Outputs Delving into model predictions helps us understand how our model outperforms the others. We examined erroneous 1-best predictions on the Ro–En data. We provide salient examples of phenomena we identified in Table 5. (Naturally, as the Ro–En score differences are not dramatic, the predictions are largely similar.)

Several examples support the fact that our model has more fluent and accurate translations than the baseline or VNMT. VNMT often struggles by introducing disfluent words, and both VNMT and Transformer select justifiable but incorrect words. For instance, in our second example, the gender and animacy of the possessor are not specified in Romanian. Our model selects a more plausible pronoun for this context.

Analysis of Latent Variables Finally, we probe whether different latent variables encode different information. We random sample 100 sentences from two test sets of distinct domains, MTNT (Reddit comments) and WMT (news) with 50 sentences each. We plot the t-SNE projection of their corresponding samples $\boldsymbol{z}_k$ inferred from Φ_k, $k = 1, 2, 3, 4$ respectively. Figure 4 suggests that different latent variables learn to organize the data in different manners, but there was no clear signal that any of them exclusively specialize in encoding a domain label. We leave a thorough analysis of

Source: ma intristeaza foarte tare .	
Reference: that really saddens me .	
Base: i am very saddened .	
VNMT: i am saddened very loudly .	*(Wrong sense of* tare*)*
Ours: i am very saddened .	
Source: cred ca executia sa este gresita .	
Reference: i believe his execution is wrong .	
Base: i believe that its execution is wrong .	
VNMT: i believe that its execution is wrong .	
Ours: i believe that his execution is wrong .	
Source: da , chinatown	
Reference: yes , chinatown	
Base: yes , chinatown	
VNMT: yes , thin .	
Ours: yes , chinatown	
Source: nu stiu cine va fi propus pentru aceasta functie .	
Reference: i do not know who will be proposed for this position .	
Base: i do not know who will be proposed for this function .	
VNMT: i do not know who will be proposed for this function .	
Ours: i do not know who will be proposed for this position .	
Source: recrutarea , o prioritate tot mai mare pentru companii	
Reference: recruitment , a growing priority for companies	
Base: recruitment , an increasing priority for companies	
VNMT: recruitment , [article missing] increasing priority for companies	
Ours: recruitment , a growing priority for companies	

Table 5: Translation examples from the baseline Transformer, VNMT, and our model. Disfluent words or absences are in red, and slightly incorrect lexical choice is in blue. Romanian diacritics have been stripped.

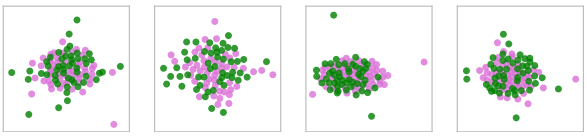

Figure 4: t-SNE visualization of $\boldsymbol{z}_k$, $k = 1, 2, 3, 4$ samples from 100 sentences from two datasets with distinct domains, MTNT (orchid) and WMT news (green).

their information specialization to future work.

7 Related Work

Unlike most prior work in (conditional) text generation, we tackle posterior collapse without requiring an annealing schedule (Bowman et al., 2016; Sønderby et al., 2016; Kim et al., 2018), a weakened decoder (Gulrajani et al., 2017), or a restricted variational family (Razavi et al., 2019).

Unlike Ma et al. (2018), who also employ bag-of-words as an NMT objective, our BoW decoder only sees the latent variable $\boldsymbol{z}$, not the encoder states. Conversely, unlike Weng et al. (2017), our generative decoder has access to both the latent variable and the encoder states; bag-of-words prediction is handled by separate parameters.

VNMT (Zhang et al., 2016) applies CVAE with Gaussian priors to conditional text generation. VRNMT (Su et al., 2018) extends VNMT, modeling the translation process in greater granularity. Both needed manually designed annealing schedules to increase KL loss and avoid posterior collapse. Discrete latent variables have been applied to NMT (Kaiser et al., 2017; Gu et al., 2018; Shen et al., 2019), without variational inference or addressing posterior collapse. Approaches to stop posterior collapse include aggressively trained inference networks (He et al., 2019), skip connections (Dieng et al., 2019), and expressive priors (Tomczak and Welling, 2018; Razavi et al., 2019).

Unlike our conditional approach, Shah and Barber (2018) jointly model the source and target text in a generative fashion. Their EM-based inference is more computationally expensive than our amortized variational inference. Eikema and Aziz (2019) also present a generative (joint) model relying on autoencoding; they condition the source text $\boldsymbol{x}$ on the latent variable $\boldsymbol{z}$. Finally, Schulz et al. (2018), like us, value mutual information between the data and the latent variable. While they motivate KL annealing using mutual information, we show that the annealing is unnecessary.

8 Conclusion

We have presented a conditional generative model with latent variables whose distribution is learned with variation inference, then evaluated it in machine translation. Our approach does not require an annealing schedule or a hamstrung decoder to avoid posterior collapse. Instead, by providing a new analysis of the conditional VAE objective to improve it in a principled way and incorporating an auxiliary decoding objective, we measurably prevented posterior collapse.

As a result, our model has outperformed previous variational NMT models in terms of translation quality, and is comparable to non-latent Transformer on standard WMT Ro↔En and De↔En datasets. Furthermore, the proposed method has improved robustness in dealing with uncertainty in data, including exploiting source-side monolingual data as well as training with noisy parallel data.

9 Acknowledgments

We thank Alexandra DeLucia, Chu-Cheng Lin, Hongyuan Mei, Kenton Murray, Guanghui Qin, and João Sedoc (alphabetically) for remarks on the exposition.

References

Léon Bottou and Yann L. Cun. 2004. Large scale online learning. In *Advances in Neural Information Processing Systems 16*, pages 217–224. Curran Associates, Inc.

Samuel R. Bowman, Luke Vilnis, Oriol Vinyals, Andrew Dai, Rafal Jozefowicz, and Samy Bengio. 2016. Generating sentences from a continuous space. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 10–21, Berlin, Germany. Association for Computational Linguistics.

Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, Jennifer C. Lai, and Robert L. Mercer. 1992. An estimate of an upper bound for the entropy of English. *Computational Linguistics*, 18(1):31–40.

Xi Chen, Diederik P. Kingma, Tim Salimans, Yan Duan, Prafulla Dhariwal, John Schulman, Ilya Sutskever, and Pieter Abbeel. 2017. Variational lossy autoencoder. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*. OpenReview.net.

Yong Cheng, Wei Xu, Zhongjun He, Wei He, Hua Wu, Maosong Sun, and Yang Liu. 2016. Semi-supervised learning for neural machine translation. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1965–1974, Berlin, Germany. Association for Computational Linguistics.

Thomas M. Cover and Joy A. Thomas. 2006. *Elements of Information Theory (Wiley Series in Telecommunications and Signal Processing)*. Wiley, New York, NY, USA.

Anna Currey, Antonio Valerio Miceli Barone, and Kenneth Heafield. 2017. Copied monolingual data improves low-resource neural machine translation. In *Proceedings of the Second Conference on Machine Translation*, pages 148–156, Copenhagen, Denmark. Association for Computational Linguistics.

Adji B. Dieng, Yoon Kim, Alexander M. Rush, and David M. Blei. 2019. Avoiding latent variable collapse with generative skip models. In *Proceedings of Machine Learning Research*, volume 89 of *Proceedings of Machine Learning Research*, pages 2397–2405.

Bryan Eikema and Wilker Aziz. 2019. Auto-encoding variational neural machine translation. In *Proceedings of the 4th Workshop on Representation Learning for NLP (RepL4NLP-2019)*, pages 124–141, Florence, Italy. Association for Computational Linguistics.

Jiatao Gu, James Bradbury, Caiming Xiong, Victor O. K. Li, and Richard Socher. 2018. Non-autoregressive neural machine translation. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Ishaan Gulrajani, Kundan Kumar, Faruk Ahmed, Adrien Ali Taïga, Francesco Visin, David Vázquez, and Aaron C. Courville. 2017. 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.

Francisco Guzmán, Peng-Jen Chen, Myle Ott, Juan Pino, Guillaume Lample, Philipp Koehn, Vishrav Chaudhary, and Marc'Aurelio Ranzato. 2019. The FLORES evaluation datasets for low-resource machine translation: Nepali–English and Sinhala–English. 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)*, pages 6098–6111, Hong Kong, China. Association for Computational Linguistics.

Junxian He, Daniel Spokoyny, Graham Neubig, and Taylor Berg-Kirkpatrick. 2019. Lagging inference networks and posterior collapse in variational autoencoders. In *ICLR*.

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

Eric Jang, Shixiang Gu, and Ben Poole. 2017. Categorical reparameterization with Gumbel–Softmax. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*.

Lukasz Kaiser, Aidan N. Gomez, Noam Shazeer, Ashish Vaswani, Niki Parmar, Llion Jones, and Jakob Uszkoreit. 2017. One model to learn them all. *CoRR*, abs/1706.05137v1.

Nal Kalchbrenner and Phil Blunsom. 2013. Recurrent convolutional neural networks for discourse compositionality. In *Proceedings of the Workshop on Continuous Vector Space Models and their Compositionality*, pages 119–126, Sofia, Bulgaria. Association for Computational Linguistics.

Huda Khayrallah and Philipp Koehn. 2018. On the impact of various types of noise on neural machine translation. In *Proceedings of the 2nd Workshop on Neural Machine Translation and Generation*, pages 74–83, Melbourne, Australia. Association for Computational Linguistics.

Yoon Kim, Sam Wiseman, Andrew Miller, David Sontag, and Alexander Rush. 2018. Semi-amortized variational autoencoders. In *Proceedings of the*

35th International Conference on Machine Learning, volume 80 of *Proceedings of Machine Learning Research*, pages 2678–2687, Stockholmsmässan, Stockholm Sweden. PMLR.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

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

Adam Lopez and Matt Post. 2013. Beyond bitext: Five open problems in machine translation. In *Twenty Years of Bitext*.

Shuming Ma, Xu Sun, Yizhong Wang, and Junyang Lin. 2018. Bag-of-words as target for neural machine translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 332–338, Melbourne, Australia. Association for Computational Linguistics.

Chris J. Maddison, Andriy Mnih, and Yee Whye Teh. 2017. The concrete distribution: A continuous relaxation of discrete random variables. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*.

Paul Michel and Graham Neubig. 2018. MTNT: A testbed for machine translation of noisy text. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 543–553, Brussels, Belgium. Association for Computational Linguistics.

Radford M. Neal and Geoffrey E. Hinton. 1998. A view of the EM algorithm that justifies incremental, sparse, and other variants. In Michael I. Jordan, editor, *Learning in Graphical Models*, pages 355–368. Springer Netherlands, Dordrecht.

Myle Ott, Michael Auli, David Grangier, and Marc'Aurelio Ranzato. 2018a. Analyzing uncertainty in neural machine translation. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, volume 80 of *Proceedings of Machine Learning Research*, pages 3953–3962. PMLR.

Myle Ott, Sergey Edunov, David Grangier, and Michael Auli. 2018b. Scaling neural machine translation. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 1–9, Belgium, Brussels. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers*, pages 186–191, Belgium, Brussels. Association for Computational Linguistics.

Ali Razavi, Aäron van den Oord, Ben Poole, and Oriol Vinyals. 2019. 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.

Danilo Jimenez Rezende and Shakir Mohamed. 2015. Variational inference with normalizing flows. In *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*, volume 37 of *JMLR Workshop and Conference Proceedings*, pages 1530–1538. JMLR.org.

Philip Schulz, Wilker Aziz, and Trevor Cohn. 2018. A stochastic decoder for neural machine translation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1243–1252, Melbourne, Australia. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016a. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016b. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1715–1725, Berlin, Germany. Association for Computational Linguistics.

Harshil Shah and David Barber. 2018. Generative neural machine translation. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems 31*, pages 1346–1355. Curran Associates, Inc.

Tianxiao Shen, Myle Ott, Michael Auli, and Marc'Aurelio Ranzato. 2019. Mixture models for diverse machine translation: Tricks of the trade. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, volume 97 of *Proceedings of Machine Learning Research*, pages 5719–5728. PMLR.

Jason R. Smith, Herve Saint-Amand, Magdalena Plamada, Philipp Koehn, Chris Callison-Burch, and Adam Lopez. 2013. Dirt cheap web-scale parallel text from the common crawl. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1374–1383, Sofia, Bulgaria. Association for Computational Linguistics.

Kihyuk Sohn, Honglak Lee, and Xinchen Yan. 2015. Learning structured output representation using deep conditional generative models. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 3483–3491. Curran Associates, Inc.

Casper Kaae Sønderby, Tapani Raiko, Lars Maaløe, Søren Kaae Sønderby, and Ole Winther. 2016. Ladder variational autoencoders. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 3738–3746. Curran Associates, Inc.

Jinsong Su, Shan Wu, Deyi Xiong, Yaojie Lu, Xianpei Han, and Biao Zhang. 2018. Variational recurrent neural machine translation. In *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18), New Orleans, Louisiana, USA, February 2-7, 2018*, pages 5488–5495. AAAI Press.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 3104–3112. Curran Associates, Inc.

Christoph Tillmann and Hermann Ney. 2003. Word reordering and a dynamic programming beam search algorithm for statistical machine translation. *Computational Linguistics*, 29(1):97–133.

Jakub M. Tomczak and Max Welling. 2018. VAE with a VampPrior. In *International Conference on Artificial Intelligence and Statistics, AISTATS 2018, 9-11 April 2018, Playa Blanca, Lanzarote, Canary Islands, Spain*, volume 84 of *Proceedings of Machine Learning Research*, pages 1214–1223. PMLR.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 5998–6008. Curran Associates, Inc.

Rongxiang Weng, Shujian Huang, Zaixiang Zheng, Xinyu Dai, and Jiajun Chen. 2017. Neural machine translation with word predictions. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 136–145, Copenhagen, Denmark. Association for Computational Linguistics.

Lijun Wu, Yiren Wang, Yingce Xia, Tao QIN, Jianhuang Lai, and Tie-Yan Liu. 2019. Exploiting monolingual data at scale for neural machine translation. 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)*, pages 4205–4215, Hong Kong, China. Association for Computational Linguistics.

Hainan Xu and Philipp Koehn. 2017. Zipporah: a fast and scalable data cleaning system for noisy web-crawled parallel corpora. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2945–2950, Copenhagen, Denmark. Association for Computational Linguistics.

Yilin Yang, Liang Huang, and Mingbo Ma. 2018a. Breaking the beam search curse: A study of (re-)scoring methods and stopping criteria for neural machine translation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3054–3059, Brussels, Belgium. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Ruslan Salakhutdinov, and William W. Cohen. 2018b. Breaking the softmax bottleneck: A high-rank RNN language model. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Biao Zhang, Deyi Xiong, Jinsong Su, Qun Liu, Rongrong Ji, Hong Duan, and Min Zhang. 2016. Variational neural discourse relation recognizer. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 382–391, Austin, Texas. Association for Computational Linguistics.

Jiajun Zhang and Chengqing Zong. 2016. Exploiting source-side monolingual data in neural machine translation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1535–1545, Austin, Texas. Association for Computational Linguistics.

Shengjia Zhao, Jiaming Song, and Stefano Ermon. 2019. Infovae: Balancing learning and inference in variational autoencoders. 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 5885–5892. AAAI Press.

Tiancheng Zhao, Kyusong Lee, and Maxine Eskenazi. 2018. Unsupervised discrete sentence representation learning for interpretable neural dialog generation. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1098–1107, Melbourne, Australia. Association for Computational Linguistics.

A Derivation of Equation 5

To prove the decomposition of the conditional VAE's regularization term into a mutual information term and a KL divergence term, we introduce a random variable ℓ representing an index into the training data; it uniquely identifies $\left(\boldsymbol{x}^{(\ell)}, \boldsymbol{y}^{(\ell)}\right)$. This alteration is "entirely algebraic" (Hoffman and Johnson, 2016) while making our process both more compact and more interpretable.

$$q(\ell, \boldsymbol{z}) \triangleq q(\ell)q(\boldsymbol{z} \mid \ell) \qquad q(\boldsymbol{z} \mid \ell) \triangleq q(\boldsymbol{z} \mid \boldsymbol{x}^{(\ell)}, \boldsymbol{y}^{(\ell)}) \qquad q(\ell) \triangleq \frac{1}{L}$$

$$p(\ell, \boldsymbol{z}) \triangleq p(\ell)p(\boldsymbol{z} \mid \ell) \qquad p(\boldsymbol{z} \mid \ell) \triangleq p(\boldsymbol{z}) \qquad p(\ell) \triangleq \frac{1}{L}$$

We define the marginals $p(\boldsymbol{z})$ and $q(\boldsymbol{z})$ as the aggregated posterior (Tomczak and Welling, 2018) and aggregated approximate posterior (Hoffman and Johnson, 2016). (This allows the independence assumption above.) Moving forward will require just a bit of information theory: the definitions of entropy and mutual information. For these, we direct the reader to the text of Cover and Thomas (2006).

Given these definitions, the regularization term of the ELBO objective may be expressed as

$$\mathbb{E}_\ell \left[\mathrm{D}_{\mathrm{KL}} \left(q(\boldsymbol{z} \mid \boldsymbol{x}, \boldsymbol{y}) \parallel p(\boldsymbol{z} \mid \boldsymbol{x})\right)\right] = \sum_\ell \frac{1}{L} q(\boldsymbol{z} \mid \boldsymbol{x}, \boldsymbol{y}) \log \frac{q(\boldsymbol{z} \mid \boldsymbol{x}, \boldsymbol{y})}{p(\boldsymbol{z} \mid \boldsymbol{x})}.$$

We may now multiply the numerator and denominator by $\frac{1}{L}$ and use its equivalence to $p(\ell)$ and $q(\ell)$.

$$= \sum_\ell q(\ell, \boldsymbol{z}) \log \frac{q(\ell, \boldsymbol{z})}{p(\ell, \boldsymbol{z})}$$

Factoring then gives us two $\log$ terms.

$$= \sum_\ell q(\ell, \boldsymbol{z}) \left[\log \frac{q(\boldsymbol{z})}{p(\boldsymbol{z})} + \log \frac{q(\ell \mid \boldsymbol{z})}{p(\ell)}\right]$$

We then distribute the weighted sum.

$$= \mathrm{D}_{\mathrm{KL}}(q(\boldsymbol{z}) \parallel p(\boldsymbol{z})) + \mathbb{E}_{q(\boldsymbol{z})} \left[\mathrm{D}_{\mathrm{KL}}(q(\ell \mid \boldsymbol{z}) \mid p(\ell))\right]$$

Because of how we defined $p(\ell)$, we expand the second term and factor out the constant $\mathrm{H}(p(\ell)) = \log L$.

$$= \mathrm{D}_{\mathrm{KL}}(q(\boldsymbol{z}) \parallel p(\boldsymbol{z})) + \log L - \mathbb{E}_{q(\boldsymbol{z})} \left[\mathrm{H}(q(\ell \mid \boldsymbol{z}))\right]$$

Finally, we arrive at the result from Equation 5 by using $\log L = \mathrm{H}(q(\ell))$.

$$= \mathrm{D}_{\mathrm{KL}}(q(\boldsymbol{z}) \parallel p(\boldsymbol{z})) + \mathrm{I}_q(\ell; \boldsymbol{z}).$$

□