Discrete Auto-regressive Variational Attention Models for Text Modeling

Xianghong Fang*†, Haoli Bai*†, Jian Li†, Zenglin Xu‡, Michael Lyu†, Irwin King†

†Department of Computer Science and Engineering, The Chinese University of Hong Kong
‡School of Computer Science and Engineering, Harbin Institute of Technology, Shenzhen
Email: xianghong_fang@163.com, {hlbai, jianli, lyu, king}@cse.cuhk.edu.hk, xuzenglin@hit.edu.cn
*: Equal Contribution

Abstract—**Variational autoencoders (VAEs) have been widely applied for text modeling. In practice, however, they are troubled by two challenges: information underrepresentation and posterior collapse. The former arises as only the last hidden state of LSTM encoder is transformed into the latent space, which is generally insufficient to summarize the data. The latter is a long-standing problem during the training of VAEs as the optimization is trapped to a disastrous local optimum. In this paper, we propose Discrete Auto-regressive Variational Attention Model (DAVAM) to address the challenges. Specifically, we introduce an auto-regressive variational attention approach to enrich the latent space by effectively capturing the semantic dependency from the input. We further design discrete latent space for the variational attention and mathematically show that our model is free from posterior collapse. Extensive experiments on language modeling tasks demonstrate the superiority of DAVAM against several VAE counterparts. Code will be released.**

Index Terms—**Text Modeling, Information Underrepresentation, Posterior Collapse**

I. INTRODUCTION

As one of the representative deep generative models, variational autoencoders (VAEs) [1] have been widely applied in text modeling [2]–[6]. Given input text $\mathbf{x} \in \mathcal{X}$, VAEs learn the variational posterior $q_\phi(\mathbf{z}|\mathbf{x})$ through the encoder and reconstruct output $\hat{\mathbf{x}}$ from latent variables $\mathbf{z}$ via the decoder $p_\theta(\mathbf{x}|\mathbf{z})$. Both encoder and decoder are usually implemented by deep recurrent networks such as LSTMs [7] in text modeling. Despite the success of VAEs, two long-standing challenges exist for such variational models: information underrepresentation and posterior collapse.

The challenge of information underrepresentation refers to the limited expressiveness of the latent space $\mathbf{z}$. As shown in the left of Figure 1, current VAEs build a single latent variable $\mathbf{z} = z_T$ based on the last hidden state of LSTM encoder [5], [6], [8], [9]. However, this is generally insufficient to summarize the input sentence [10]. Thus the generated sentences from the decoder are often poorly correlated. Notably, the sequence of encoder hidden states reflects the semantic dependency of the input sentence, and the whole hidden context may benefit the generation. Therefore, a potential solution is to enhance the representation power of VAEs via the attention mechanism [11], [12], a superior component in discriminative models. However, the attention module cannot be directly deployed in generative models like VAEs, as the attentional context vectors are hard
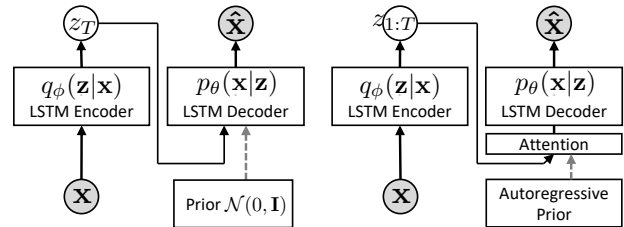

Fig. 1. Illustration of conventional VAEs (left) and our proposed auto-regressive variational attention models (right).

to compute from randomly sampled latent variables during the generation phase.

Posterior collapse is another well-known problem during the training of VAEs [13]. It occurs as the variational posterior $q_\phi(\mathbf{z}|\mathbf{x})$ converges to the prior distribution $p(\mathbf{z})$, thus the decoder receives no supervision from the input $\mathbf{x}$. Previous efforts alleviate this issue by either annealing the KL divergence term [8], [13], [14], revising the model [15]–[17], or modifying the training procedure [6], [9]. Nevertheless, they primarily focus on a single latent variable for language modeling, which still suffer from the information underrepresentation as mentioned before. To derive more powerful latent space, the challenge of posterior collapse should be carefully handled.

In this paper, we propose Discrete Auto-regressive Variational Attention Model (DAVAM) to address the aforementioned challenges. First, to mitigate the information underrepresentation of VAEs, we introduce a variational attention mechanism together with an auto-regressive prior (dubbed as *auto-regressive variational attention*). The variational attention assigns a latent sequence $\mathbf{z} = z_{1:T}$ over each encoder hidden state to capture the semantic dependency from the input, as is shown in the right of Figure 1. During the generation phase, the auto-regressive prior generates well-correlated latent sequence for computing the attentional context vectors. Second, we utilize *discrete latent space* to tackle the posterior collapse in VAEs. We show that the proposed auto-regressive variational attention models, when armed with conventional Gaussian distribution, face high risks of posterior collapse. Inspired by the recently proposed Vector Quantized Variational Autoencoder (VQ-VAE) [18], [19], we design a discrete latent distribution over the variational attention mechanism. By analyzing the intrinsic

merits of discreteness, we demonstrate that our design is free from posterior collapse regardless of latent sequences length. Consequently, the representation power of DAVAM can be significantly enhanced without posterior collapse.

We evaluate DAVAM on several benchmark datasets on language modeling. The experimental results demonstrate the superiority of our proposed method in text generation over its counterparts.

Our contributions can thus be summarized as:

1) To the best of our knowledge, this is the first work that proposes *auto-regressive variational attention* to improve VAEs for text modeling, which significantly enriches the information representation of latent space.

2) We further design *discrete latent space* for the proposed variational attention, which effectively addresses the posterior collapse issue during the optimization.

II. BACKGROUND

A. Variational Antoencoders for Text Modeling

Variational Autoencoders (VAEs) [1] are a well known class of generative models. Given sentences $\mathbf{x} = x_{1:T}$ with length T, we seek to infer latent variables $\mathbf{z}$ that explain the observation. To achieve this, we need to maximize the marginal log-likelihood $\log p_\theta(\mathbf{x})$, which is usually intractable due to the complex posterior $p(\mathbf{z}|\mathbf{x})$. Consequently an approximate posterior $q_\phi(\mathbf{z}|\mathbf{x})$ (i.e. the *encoder*) is introduced, and the evidence lower bound (ELBO) of the marginal likelihood is maximized as follows:

$$
\log p_\theta(\mathbf{x}) \geq \underbrace{\mathbb{E}_{\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})]}_{\text{reconstruction loss}}
$$
$$
- \underbrace{D_{KL}(q_\phi(\mathbf{z}|\mathbf{x})\|p(\mathbf{z}))}_{\text{KL divergence}}, \quad (1)
$$

where $p_\theta(\mathbf{x}|\mathbf{z})$ represents likelihood function conditioned on $\mathbf{z}$, also known as the *decoder*. In the context of text modeling, both encoder and decoder are usually implemented by deep recurrent models such as LSTMs [7], parameterized by ϕ and θ respectively.

B. Challenges

a) Information Underrepresentation: Information underrepresentation is a common issue in applying VAEs for text modeling. Conventional VAEs build latent variables based on the last hidden state of LSTM encoder, i.e. $\mathbf{z} = z_T$. During the decoding process, we first sample z_T, from which new sentences $\hat{\mathbf{x}} = \hat{x}_{1:\hat{T}}$ can be generated:

$$
p(\hat{\mathbf{x}}|\mathbf{z}) = p_\theta(\hat{x}_1|z_T) \prod_{t=2}^{\hat{T}} p_\theta(\hat{x}_t|\hat{x}_{t-1}, z_T), \quad (2)
$$

where $\hat{T}$ is the length of reconstructed sentence $\hat{\mathbf{x}}$. However, the representation of z_T is generally insufficient to summarize the semantic dependencies in $\mathbf{x}$, and thus deteriorates the reconstruction.

b) Posterior Collapse: Posterior collapse usually arises as $D_{KL}(q_\phi(\mathbf{z}|\mathbf{x})\|p(\mathbf{z}))$ diminishes to zero, where the local optimal gives $q_\phi(\mathbf{z}|\mathbf{x}) = p(\mathbf{z})$. Posterior collapse happens inevitably as the ELBO contains both the reconstruction loss $\mathbb{E}_{\mathbf{z} \sim q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})]$ and the KL-divergence $D_{KL}(q_\phi(\mathbf{z}|\mathbf{x})\|p(\mathbf{z}))$, as shown in Equation (1). When posterior collapse happens, $\mathbf{x}$ becomes independent of $\mathbf{z}$ as $p(\mathbf{x})p(\mathbf{z}) = p(\mathbf{x})q_\phi(\mathbf{z}|\mathbf{x}) = p(\mathbf{x})\frac{p(\mathbf{x},\mathbf{z})}{p(\mathbf{x})} = p(\mathbf{x},\mathbf{z})$. Therefore, the encoder learns a data-agnostic posterior without any information from $\mathbf{x}$, while the decoder fails to perform valid generation but purely based on random noise.

III. METHODOLOGY

We now present our solutions to address the aforementioned challenges. In order to enrich the latent space, we propose an auto-regressive variational attention model to capture the semantic dependencies in the input space. We first instantiate variational attention with the Gaussian distribution and show that it suffers from posterior collapse. Then to solve the challenge, we further discretize the latent space with one-hot categorical distribution, leading to discrete auto-regressive variational attention models (DAVAM), as illustrated in Figure 2. We carefully analyze the superiority of DAVAM to avoid posterior collapse.

A. Gaussian Auto-regressive Variational Attention Models

To enrich the representation of latent space $\mathbf{z}$, we seek to incorporate the attention mechanism into VAEs. Specifically, we denote the encoder hidden states as $h_{1:T}^e$, and the decoder hidden states as $h_{1:\hat{T}}^d$. We build a latent sequence $\mathbf{z} = z_{1:T}$ upon encoder hidden states $h_{1:T}^e$. To facilitate such variational attention model, one can choose the conventional Gaussian distribution [1] for variational posteriors, i.e. $q(\mathbf{z}|\mathbf{x}) = \prod_{t=1}^{T} q(z_t|\mathbf{x})$ where $q_\phi(z_t|\mathbf{x}) = \mathcal{N}(\mu_t, \sigma_t I)$. We name the resulting model as Gaussian Auto-regressive Variational Attention Model (GAVAM).

Given $z_{1:T}$, similar to attention-based sequence-to-sequence (seq2seq) models [11], the attentional context vectors c_i and scores at i-th decoding step can be computed by

$$
c_i = \sum_{t=1}^{T} \alpha_{i,t} z_t, \quad \alpha_{i,t} = \frac{\exp(\tilde{\alpha}_{i,j})}{\sum_{j=1}^{T} \exp(\tilde{\alpha}_{i,j})}, \quad (3)
$$

where $\tilde{\alpha}_{i,t} = v^\top \tanh(W_e z_t + W_d h_{i-1}^d + b)$ is the unnormalized score, $t \in \{1, 2, ..., T\}$ is the encoder time step, and W_e, W_d are the corresponding parameters. By taking c_i as extra input to the decoder, the generation process is reformulated as

$$
p(\hat{\mathbf{x}}|\mathbf{c}, \mathbf{z}) = p(\hat{x}_1|c_1, z_{1:T}) \prod_{\hat{t}=2}^{\hat{T}} p(\hat{x}_{\hat{t}}|\hat{x}_{\hat{t}-1}, c_{\hat{t}}, z_{1:T}).
$$

Unlike Equation 2, at each time step, the decoder receives supervision from the context vector, which is a weighted sum of the latent sequence $z_{1:T}$. Consequently, the variational posterior $q_\phi(z_{1:T}|\mathbf{x})$ encodes the semantic dependency from the observations, such that the issue of information underrepresentation can be effectively mitigated.
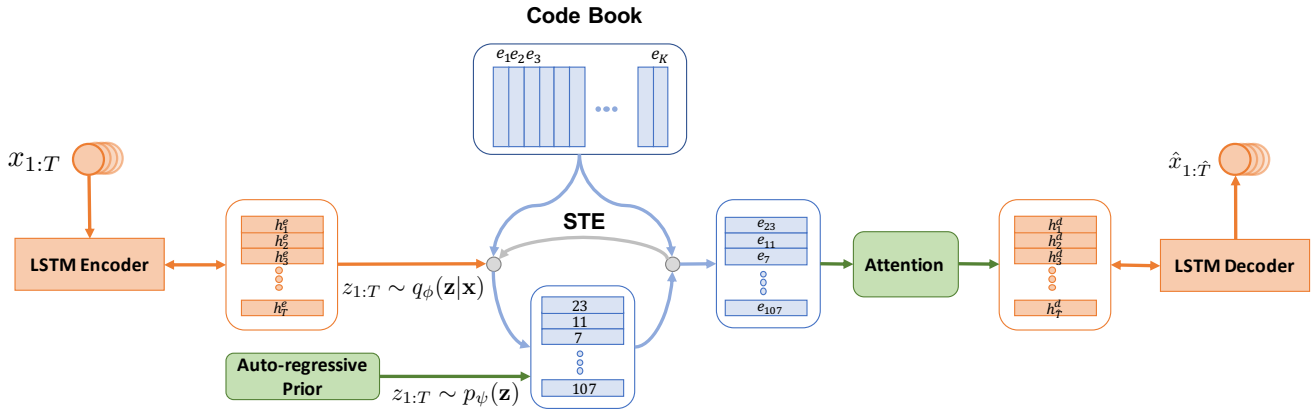
Fig. 2. The overall architecture of the proposed DAVAM. Given observations $\mathbf{x} = x_{1:T}$, the encoder hidden states $h^e_{1:T}$ are quantized to code book $\{e_k\}^K_{k=1}$ based on index sequence $z_{1:T}$ from the posterior. The quantized hidden states $e_{z_{1:T}}$ are then forwarded to the attention module with decoder hidden states $h^d_{1:\hat{T}}$. During back-propagation, the gradients of $e_{z_{1:T}}$ are directly copied to $h^e_{1:T}$ with STE. To generate new sentences from DAVAM, we start from the auto-regressive prior to sample a new latent sequence $z_{1:T}$. The sequence $z_{1:T}$ is then used to index the code book for attention computation during decoding.

a) Auto-regressive Prior: A key difference between variational auto-regressive attention models and conventional VAEs is the choice of a prior distribution. During the generation, the latent sequence $z_{1:T}$ are sampled from the prior unconditionally, and are then fed to the attention module together with $h^d_{1:\hat{T}}$. The most adopted prior $\mathcal{N}(0, I)$, however, is non-informative to generate *well-correlated* latent sequence for the attention as that during training. Therefore the decoder receives no informative supervision that gives reasonable generation.

To solve that, we deploy an auto-regressive prior $p_\psi(z_{1:T}) = p_\psi(z_1)\prod_{t=2}^T p_\psi(z_t|z_{1:t-1})$ parameterized by ψ to capture the underlying semantic dependencies. Specifically, we take $p_\psi(z_t|z_{1:t-1}) = \mathcal{N}(\hat{\mu}_t, \hat{\sigma}_t I)$, where $(\hat{\mu}_t, \hat{\sigma}_t I)$ is produced by a PixelCNN, a superior model in learning sequential data [20].

b) Posterior Collapse in GAVAM: The training of GAVAM can be easily troubled by posterior collapse due to two aspects. To see this, similar to Equation 1, the minimization of the ELBO now can be written as:

$$\min_{\phi,\theta,\psi} - \mathbb{E}_{z_{1:T}\sim q_\phi}[\log p_\theta(\mathbf{x}|z_{1:T})] \qquad (4)$$

$$+ \sum_{t=1}^T D_{KL}(q_\phi(z_t|\mathbf{x})\|p_\psi(z_t|z_{1:t-1})).$$

On the one hand, the KL divergence scales linearly to the sequence length of T, which makes the training unstable across different input lengths. On the other hand, and more seriously, both ϕ and ψ are used to minimize the KL divergence, which can easily trap the learned posteriors. To demonstrate this, for example, the KL divergence between two Gaussian distributions can be written as:

$$\sum_{t=1}^T D_{KL}(q_\phi(z_t|\mathbf{x})\|p_\psi(z_t|z_{1:t-1})) \qquad (5)$$

$$= \sum_{t=1}^T \sum_{d=1}^D \frac{1}{2}(\log\frac{\hat{\sigma}^2_{td}}{\sigma^2_{td}} - 1 + \frac{\sigma^2_{td} + (\hat{\mu}_{td} - \mu_{td})^2}{\hat{\sigma}^2_{td}}),$$

where D is the latent dimension of z_t. Whenever $\sigma^2_{td} \to \hat{\sigma}^2_{td}$ and $\mu_{td} \to \hat{\mu}_{td}$ before $q_\phi(z_{1:T}|\mathbf{x})$ encodes anything from $\mathbf{x}$, both $q_\phi(z_{1:T}|\mathbf{x})$ and $p_\psi(z_{1:T})$ get stuck in local optimal and learn no semantic dependency for reconstruction.

B. Discrete Auto-regressive Variational Attention Models

Inspired by recent studies [18], [21] that demonstrate the promising effects of *discrete latent space*, we explore its potential in handling posterior collapse over the variational attention, leading to discrete auto-regressive variational attention model (DAVAM).

Specifically, we introduce a code book $\{e_k\}^K_{k=1}$ with size of K, where each e_k is a vector in the latent space. We expect the combination of code book can represent the semantic dependency from observed sentence $\mathbf{x}$. We now substitute the Gaussian distributed $z_{1:T}$ with discrete indices over code book that follows one-hot categorical distribution:

$$q_\phi(z_t = k|\mathbf{x}) = \begin{cases} 1 & k = \arg\min_j \|h^e_t - e_j\|_2 \\ 0 & \text{otherwise} \end{cases}. \qquad (6)$$

Given index z_t, we transform the the encoder hidden state h^e_t to the nearest e_{z_t}. Then we use e_{z_t} instead of z_t in Equation 3 to compute attention scores α_t and the context vectors c_i.

Correspondingly, as $z_{1:T}$ are discrete indices, we assign categorical distribution for the auto-regressive prior, i.e., $p_\psi(z_t|z_{1:t-1}) = \text{Cat}(\gamma_t)$. The categorical parameter can be obtained from the PixelCNN model given historical records $z_{1:t-1}$, i.e, $\gamma_t = \text{PixelCNN}_\psi(z_{1:t-1}) \in [0, 1]^K$.

a) Advantages of Discreteness: Thanks to the nice properties of discreteness, the optimization of DAVAM does not suffer from posterior collapse. Specifically, the KL divergence

of DAVAM can be written as:

$$\sum_{t=1}^{T} D_{KL}(q_\phi(z_t|\mathbf{x})||p_\psi(z_t|z_{1:t-1})) \tag{7}$$

$$= -\sum_{t=1}^{T}\left[H(q_\phi(z_t)) + \sum_{k=1}^{K} 1_{(z_t=k)}\log\gamma_{t,k}\right]$$

$$= -\sum_{t=1}^{T}\left[0 + \log\gamma_{t,z_t}\right],$$

where the third line is obtained with the entropy $H(q_\phi(z_t)) = -1\log 1 - 0\log 0 \to 0$. It can be found that $D_{KL}(q_\phi(z_{1:T}|\mathbf{x})||p_\psi(z_{1:T}))$ is no longer relevant to posterior parameters ϕ. Consequently, the update of variational posterior $q_\phi(z_{1:T}|\mathbf{x})$ does not rely on the prior but is determined purely by the reconstruction term. Therefore minimization of KL divergence will not lead to posterior collapse.

C. Model Training

We first train the variational posterior $q_\phi(\mathbf{z}|\mathbf{x})$ to convergence when the latent sequence $z_{1:T}$ effectively captures the semantic dependency from input $\mathbf{x}$. Then we train the auto-regressive prior $p_\psi(\mathbf{z})$ to mimic the learned posterior, to facilitate well-correlated sequence during generation. Therefore, the training of the proposed DAVAM involves two stages, described in detail as follows:

a) Stage one: We follow the standard paradigm to minimize the ELBO of DAVAM. As shown in Equation 7, since KL divergence is neither relevant to ϕ nor θ, only the reconstruction term should be concerned. In the meanwhile, as the latent variables $z_{1:T}$ are determined based on Euclidean distances between $h_{1:T}^e$ and code book $\{e_k\}_{k=1}^{K}$, we further regularize them to stay close via a Frobenius norm. The training objective for stage one is

$$\min_{\theta,\phi} -\mathbb{E}_{q_\phi}[\log p_\theta(\mathbf{x}|z_{1:T})] + \beta\sum_{t=1}^{T}\|h_t^e - \mathrm{sg}(e)\|_F^2, \tag{8}$$

where β is the regularizer, and $\mathrm{sg}(\cdot)$ stands for stop-gradient operation. Note that the quantization in Equation 6 is non-differentiable. To allow the back-propagation algorithm to proceed, we adopt the widely employed straight through estimator (STE) [22] to copy gradients from e_{z_t} to h_t^e, as is shown in Figure 2.

For the update of code book $\{e_k\}_{k=1}^{K}$, we first apply K-means algorithm to calculate the average over all latent variables $h_{1:T}^e$ that are closest to $\{e_k\}_{k=1}^{K}$, and then take exponential moving average over the mini-batch update of the code book so as to stabilize the optimization.

b) Stage Two: After the convergence of DAVAM, we resort to update the auto-regressive prior $p_\psi(z_t|z_{1:t-1})$. To mimic the semantic dependency in the learned posterior $q_\phi(z_{1:T}|\mathbf{x})$, the prior is supposed to fit the latent sequence $z_{1:T} \sim q_\phi(z_t|\mathbf{x})$. This can be realized by the minimizing their KL-divergence w.r.t. ψ as

$$\min_\psi \sum_t D_{KL}(q_\phi(z_t|\mathbf{x})||p_\psi(z_t|z_{1:t-1})), \tag{9}$$

which can be simplified to the cross-entropy loss between $z_{1:T}$ and $\gamma_{1:T}$ according to Equation (7).

IV. EXPERIMENTS

We verify advantages of the proposed DAVAM on language modeling tasks, and testify how well can it generate sentences from random noise. Finally, we conduct a set of further analysis to shed more light on DAVAM.

A. Experimental Setup

We take three benchmark datasets of language modeling for verification: Yahoo Answers [17], Penn Tree [23], and a down-sampled version of SNLI [24]. A summary of dataset statistics is shown in Table I.

TABLE I
DATASET STATISTICS.

Datasets	Train Size	Val Size	Test Size	Avg Len
Yahoo	100,000	10,000	10,000	78.7
PTB	42,068	3,370	3,761	23.1
SNLI	100,000	10,000	10,000	9.7

a) Baselines: We compare the proposed DAVAM against a number of baselines, including the classical LSTM-based Language Modeling-(LSTM-LM), vanilla VAE [1], and its advanced variants: annealing VAE [13], cyclic annealing VAE[1] [8], lagging VAE[2] [9], Free Bits (FB) [14] and pretraining+FBP VAE[3] [6]. All these baselines do not use the attention module in their architectures.

For ablation studies, we further compare to 1) GAVAM, which takes Gaussian distribution instead of the one-hot categorical distribution over $z_{1:T}$ to verify the advantages of discreteness; 2) We also remove the attention mechanism (denoted as DAVAM-q) to test the effect of discreteness on the last latent variable z_T. 3) Finally, to check the choice of prior, we replace the auto-regressive prior with uninformative Gaussian priors, which leads to variational attention models (VAE+Attn) first proposed by bahuleyan2017variational bahuleyan2017variational.

b) Evaluation Metrics: We evaluate language modeling using three metrics: 1) Reconstruction loss (Rec) $\mathbb{E}_{\mathbf{z}\sim q_\phi(\mathbf{z}|\mathbf{x})}[\log p_\theta(\mathbf{x}|\mathbf{z})]$ that measures the ability to recover data from latent space; 2) Perplexity (PPL) measuring the capacity of language modeling; Both lower Rec and PPL give better models in general; and 3) KL divergence (KL) indicating whether posterior collapse occurs.

c) Implementation: For baselines, we keep the same hyper-parameter settings to pretraining+FBP VAE [6], e.g., the dimension of latent space, word embeddings and hidden states of LSTM. Since our latent variables are discrete, we cannot use importance weighted samples to approximate the reconstruction loss in Lagging VAE and pretraining+FBP VAE.

[1]https://github.com/haofuml/cyclical_annealing
[2]https://github.com/jxhe/vae-lagging-encoder
[3]https://github.com/bohanli/vae-pretraining-encoder

#	Methods	Yahoo			PTB			SNLI		
		Rec↓	PPL↓	KL	Rec↓	PPL↓	KL	Rec↓	PPL↓	KL
1	LSTM-LM	-	60.75	-	-	100.47	-	-	21.44	-
2	VAE	329.10	61.52	0.00	101.27	101.39	0.00	33.08	21.67	0.04
3	+anneal	328.80	61.21	0.00	101.28	101.40	0.00	31.66	21.50	1.42
4	+cyclic	333.80	66.93	2.83	101.85	108.97	1.37	30.69	23.67	3.63
5	+aggressive	322.70	59.77	5.70	100.26	99.83	0.93	31.53	21.16	1.42
6	+FBP	322.91	62.59	9.08	98.52	99.62	2.95	25.26	22.05	8.99
7	+pretraining+FBP	315.09	59.60	15.49	96.91	96.17	4.99	22.30	22.33	13.40
8	GAVAM	350.14	79.28	0.00	102.20	105.94	0.00	30.90	17.68	0.38
9	DAVAM-q (K=512)	323.10	57.14	<u>0.33</u>	95.83	79.24	<u>0.27</u>	28.16	13.71	<u>0.12</u>
10	DAVAM (K=128)	303.65	45.07	<u>1.88</u>	83.57	50.15	<u>2.23</u>	16.11	5.58	<u>2.38</u>
11	DAVAM (K=512)	**259.68**	**26.61**	<u>2.60</u>	**60.16**	**17.94**	<u>3.12</u>	**10.85**	**3.52**	<u>2.69</u>

For DAVAM and its ablation counterparts, we keep the same set of hyper-parameters. By default, we set the code-book size K as 512. We first warm up the training for 10 epochs, and then gradually increase β in Equation (8) from 0.1 to $\beta_{max} = 5.0$, in a similar spirit to annealing VAE [13]. For all experiments, we use the SGD optimizer with initial learning rate 1.0, and decay it until five counts if the validation loss does not decrease for 2 epochs. For the auto-regressive prior, we use a 16-layer PixelCNN with one-dimensional convolution followed by residual connections.

B. Experimental Results

1) Language Modeling: To compare the representation of latent space, we first perform language modeling over the testing corpus of benchmark datasets, as shown in Table II. Generally, the better the representation, the lower the Rec and PPL on observations. For DAVAM and GAVAM, we average the KL divergence along the latent sequence to make them comparable to baselines that only have one latent variable.

a) Main Results: (Rows 1-7,10-11) Comparing to baselines without variational attention, we find that our DAVAM achieves significantly better results on all three datasets, especially with larger code book size K. For example, comparing to pretraining+FBP in row 7 on Yahoo Answers dataset, DAVAM with $K = 512$ significantly reduces the reconstruction loss by 55.41, and PPL is decreased by more than a half. Therefore, our DAVAM is more expressive to summarize observations comparing to baselines without attention modules. The success verifies that DAVAM can significantly enrich the latent representation of language modeling.

In terms of posterior collapse, both vanilla VAE and some variants suffer from this issue severely as the KL of Gaussian distribution diminishes nearly to 0. However, the KL of DAVAM does not indicate posterior collapse, but only reflects how well the auto-regressive prior mimic the posterior.

b) Ablation Studies: (Rows 8-11) GAVAM performs less competitively on language modeling comparing to DAVAM. Moreover, its KL divergences are near or equal to 0. This

leads to the posterior collapse and explains why it has the sub-optimal performance with the one-hot categorical distribution substituted. In terms of DAVAM-q, it has no attention module and only learns the variational posterior with the last z_T, which naturally yields less competitive results against attention-based models. However, DAVAM-q still outperforms a number of variants of VAE, as the posterior is free from the collapse thanks to discreteness.

C. Language Generation From Scratch

In this section, we dive into the ability of language generation from scratch, i.e. generating sentences directly from random noises. We study the generated languages from two perspective: quality and diversity. Then we apply such input-free generation approach to data augmentation, which can be applied to improve the language models trained over limited corpus.

1) Generation Quality: We first visualize the sentences generated by Pretraining+FBP VAE, VAE+Attn, GAVAM, and DAVAM, along with their fluency scores (PPL). We adopt a pre-trained GPT-2 [25] [4] as the fluency evaluator, which takes the generated sentences as input and returns the corresponding perplexity scores (PPL).

From Table III, we find that VAE+Attn can hardly generate well-correlated latent sequences from uninformative Gaussian distributions as a result of non-autoregressive prior. On the other hand, GAVAM is armed with the auto-regressive prior and shows more readable sentences. Nevertheless, it suffers from poor semantic meanings due to posterior collapse, as previously shown in Table II. Finally, DAVAM can produce sentences with interpretable meanings and better fluency scores, even when the sequence length is long. This suggests the discrete latent sequence combined with auto-regressive prior enjoys unique advantages in language generation from scratch.

To study the quality more rigorously, we further sample 100 sentences randomly from VAE+Attn, GAVAM, and DAVAM with different lengths, and compare their average perplexity scores with the GPT-2 evaluator. As it is an inevitable trade-off between language modeling and generation, we also report the

[4]https://huggingface.co/transformers/model_doc/gpt2.html

TABLE III
SAMPLED SHORT, MEDIUM AND LONG SENTENCES AS WELL AS THEIR GPT-2 PPL SCORES FOR MEASURING FLUENCY.

Methods	Samples	PPL↓
pretraining +FBP VAE	• [s] i can say what can be the least in terms also in any form of power stream [/s]	5.97
	• [s] how you define yourself ? birth control. you will find yourself a dead because of your periods. [/s]	5.43
	• [s] are they allowed to join (francisco) in _UNK. giants in the first place.? check out other answers. do you miss the economy and not taking risks in the merchant form, what would you tell? go to the yahoo home page and ask what restaurants follow this one. [/s]	5.21
VAE+Attn	• [s] explain some make coming you think and represents middle line girl coming. [/s]	8.83
	• [s] who just live this in you to get usual usual help the idea for out to use guess guess thats UNK	7.16
	• [s] is masterbating masterbating anyone hi fact fact forgive forgive forgive virgin chlorine does 're hydrogen download 're whats 're does solve 're whats solve 2y germany germany monde pourquoi 'm does fun pourquoi 'm 're 'm 're solve 'm does solve pourquoi 'm 'm 'm 'm solve 'm 'm [/s]	5.01
GAVAM	• [s] generally do problems problems, do you have problems [/s]	6.57
	• [s] plz can you get a UNK envelope in e ? for me for my switched for eachother i would switched to i i think thats . i need to assume . [/s]	6.50
	• [s] what to do, yoga is there to place and pa? i can definately, but that, you will the best, but the the only amount right? the best range, it though, to do n't do to be to be out, and [/s]	5.18
DAVAM (K=512)	• [s] what is the meaning of time management? [/s]	4.52
	• [s] what should you be thankful for thanksgiving dinner and how to get some money with a thanksgiving dinner? [/s]	4.25
	• [s] is anyone willing to donate plasma if you are allergic to cancer or anything else? probably you can. i've never done any thing but it is only that dangerous to kill bacteria. i have heard that it doesn't have any effect on your immune system. [/s]	3.87

TABLE IV
GPT-2 PERPLEXITY SCORES (↓) WITH STANDARD DEVIATION (±) FOR GENERATION QUALITY (ROW 1-5), AND THE RECONSTRUCTION LOSS (↓) ON YAHOO DATASET FOR LANGUAGE MODELING (LAST ROW).

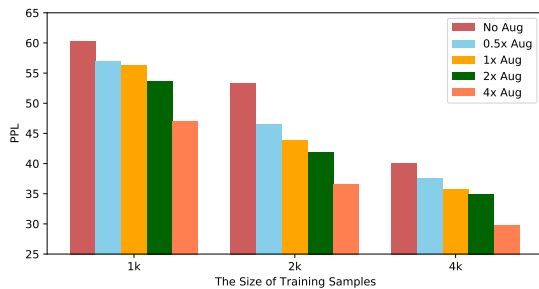| Length	VAE+Attn	GAVAM	DAVAM
10	$7.75_{\pm 1.71}$	$7.32_{\pm 1.64}$	$\mathbf{6.58_{\pm 1.17}}$
20	$7.04_{\pm 1.46}$	$6.94_{\pm 1.40}$	$\mathbf{6.49_{\pm 1.09}}$
30	$6.70_{\pm 1.31}$	$6.75_{\pm 1.38}$	$\mathbf{5.97_{\pm 0.87}}$
40	$6.54_{\pm 1.50}$	$6.28_{\pm 1.36}$	$\mathbf{5.79_{\pm 0.81}}$
50	$6.42_{\pm 1.61}$	$6.10_{\pm 1.33}$	$\mathbf{5.55_{\pm 0.97}}$
Rec	**10.85**	350.14	259.68

Fig. 3. The perplexity scores under different sizes of augmented training sentences on SNLI dataset.

reconstruction loss on Yahoo. The results are listed in Table IV. It can be found that while VAE+Attn has a superior advantage in language modeling, it has the worst GPT-2 perplexity scores since i.i.d. Gaussian noises contain no sequential information. GAVAM has minor improvement over VAE+Attn on GPT-2 scores thanks to the auto-regressive prior, but it performs poorly on language modeling due to posterior collapse. Finally, DAVAM generally achieves the lowest perplexity scores with reasonable ability in language modeling. This indicates the superiority of the auto-regressive prior for generation from scratch, and the power of discreteness to avoid posterior collapse in fitting observations.

2) Generation Diversity: Diversity is another dimension to measure the success of language generation. We follow [26] to compute the entropy and the percentage of distinct unigrams or bigrams, which are denoted as Ent., Dist-1, and Dist-2 respectively. From Table V, it can be observed that pretraining+FBP VAE achieves the highest diversity scores. However, VAE+Attn, achieves the poorest diversity scores due to repeated words, as shown in Table III. GAVAM has only minor improvement over VAE+Attn, and repeated words frequently occur as well. Finally, our DAVAM can generate diverse sentences despite the scores being slightly lower than pretraining+FBP VAE. This is due to the training of auto-regressive prior yields over-confident choices of latent codebooks, which can better capture sequential dependency with higher generation quality, but at the sacrifice of less generation diversity.

3) Generation from Scratch as Data Augmentation: Given the superior generation quality and diversity of DAVAM, we now apply it for data augmentation to improve language models that are trained over limited corpus. By amortizing the training instances into model parameters, DAVAM is able to generate sentences directly from random noises (i.e. generation from scratch). Specifically, we selectively choose the corpus size in {1k, 2k, 4k} on SNLI dataset, and use a pre-trained DAVAM model to augment $\{0.5\times, 1\times, 2\times, 4\times\}$ times of th corpus. Figure 3 shows the corresponding improvement of perplexity scores. It can be found that the perplexity decreases proportionally to the training size. Therefore, our DAVAM can be applied to language models that are trained with limited corpus for further improvement.

Length	pretraining+FBP VAE			VAE+Attn			GAVAM			DAVAM		
	Ent. ↑	Dist-1↑	Dist-2↑	Ent. ↑	Dist-1↑	Dist-2↑	Ent. ↑	Dist-1↑	Dist-2↑	Ent. ↑	Dist-1↑	Dist-2↑
10	5.41	0.412	0.853	5.09	0.366	0.792	4.78	0.288	0.724	5.00	0.366	0.844
20	5.48	0.355	0.836	5.12	0.243	0.649	4.80	0.212	0.636	5.10	0.249	0.702
30	5.57	0.301	0.798	4.70	0.166	0.487	4.60	0.150	0.503	5.18	0.210	0.655
40	5.55	0.252	0.756	4.10	0.110	0.372	4.19	0.113	0.401	5.34	0.188	0.646
50	5.69	0.249	0.765	3.92	0.089	0.326	4.02	0.093	0.354	5.33	0.173	0.611

Fig. 4. Further analyses of DAVAM.

(a) Code book size K (b) Max. regularizer β_{max} (c) Latent dimension of e_{z_t} (d) Training dynamics
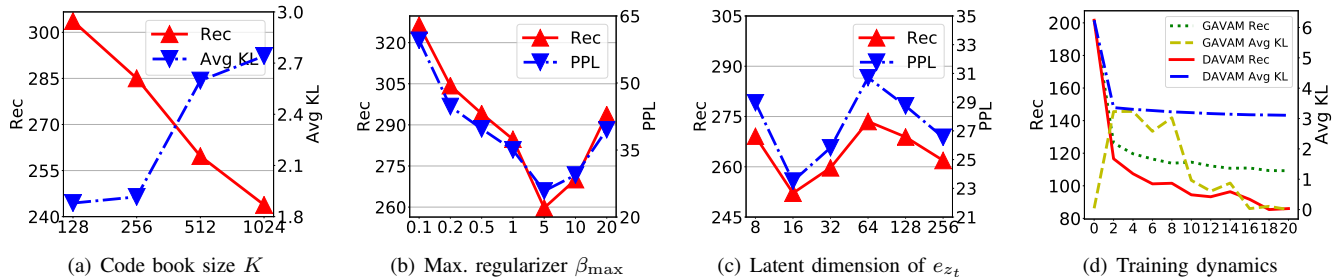
D. Further Analysis

To gain a better understanding of our proposed DAVAM, we first conduct a set of sensitivity analysis on hyper-parameter settings of the model. By default, all sensitivity analysis are conducted on Yahoo dataset with default parameter settings, except for the parameter under discussion. Then we turn to analyze training dynamics of DAVAM, which explains why DAVAM avoids posterior collapse.

1) Code Book Size K: We begin with the effect of different code book size K on the reconstruction loss and KL divergence for language modeling. We vary $K \in \{128, 256, 512, 1024\}$, and the results are shown in Figure 4(a). It can be observed that as K increases, the Rec loss decreases, whereas KL increases, both monotonically. The results are also consistent to Table II by increasing K from 128 to 512. Such phenomenons are intuitive since a larger K improves model capacity but poses more challenges for training the auto-regressive prior. Consequently, one should properly choose the code book size, such that the prior can approximate the posterior well, and yet the posterior is representative enough for the semantic dependency.

2) Maximum Regularizer β_{max}: Then we tune the maximum regularizer β_{max}, which controls the distance of the continuous hidden state $h_{1:T}^e$ to the code book $\{e_k\}_{k=1}^K$. Recall that a small β_{max} loosely restricts the continuous space $h_{1:T}^e$ to the code book, making the quantization hard to converge. On the other hand, if β_{max} is too large, $h_{1:T}^e$ could easily get stuck in some local minimal during the training. Therefore, it is necessary to find a proper trade-off between the two situations. We vary $\beta_{max} \in \{0.1, 0.2, 0.5, 1, 5, 10, 20\}$, and the results is shown in Figure 4(b). We can find that when $\beta_{max} = 5$, DAVAM achieves the lowest Rec, while smaller or larger β_{max} both lead to higher Rec values.

3) Dimension of Code Book Vectors: Finally, we change dimension of $\{e_k\}_{k=1}^K$ in $\{8, 16, 32, 64, 128, 256\}$, and the results are shown in Figure 4(c). The performance of language modeling is relatively robust to the choice of the latent dimension. This is different from the continuous space where the dimension of latent variables is closely related to the model capacity. In the discrete scenario, the model capacity is largely determined by the code book size K instead of the dimension of code book, which is also verified in Table II and Figure 4(a).

4) Training Dynamics: To empirically understand how DAVAM addresses posterior collpase, we investigate their training dynamics. We plot the curve of Rec and KL on the validation set of PTB in Figure 4(d). We find the KL of GAVAM rises at the beginning to explain observations but diminishes quickly afterward. In the meanwhile, Rec does not decrease sufficiently. This shows that the collapsed posterior fails to explain the observations. For DAVAM, on the other hand, since the optimization of reconstruction is not affected by the KL divergence, Rec is minimized sufficiently in the first place. Then we minimize KL, which converges quickly without oscillation. In other words, the posterior and prior are updated separately in two stages to avoid posterior collapse.

V. RELATED WORK

A. Variational Attention Models

Attention mechanism is commonly adopted address the issue of under-fitting in various deep generative models [10], [27], [28]. Both deng2018latent and bahuleyan2017variational consider the generation from some source input, where new latent sequences are generated conditioned on observations. Nevertheless, these methods can hardly be applied when no source information is available. Instead, our work focuses on the ability of *generation from scratch*, i.e., generating from latent space directly without external sources [29]. Generation from

scratch has various applications, such as data augmentation where new training instances can be directly generated from random noise to increase the limited training size. To enable such ability, an auto-regressive prior should be deployed to generate semantically dependent latent sequences. This explains the core idea of auto-regressive variational attention in our approach.

B. Discrete Latent Variables

Aside from the mostly used Gaussian distribution in VAEs, recent works also explore discrete latent space such as DVAE [30], DVAE++ [31], and DVAE# [32]. Nevertheless, these works have different motivations for discreteness. They introduce binary latent variables to improve the model capacity. In DAVAM, instead of enhancing the model capacity, we assign one-hot distribution on latent variables that aims to resolve posterior collapse, which is not addressed in these previous efforts [30]–[32].

VI. CONCLUSION

In this paper, we propose the discrete auto-regressive variational attention model, a new deep generative model for text modeling. The proposed approach addresses two important issues: information underrepresentation and posterior collapse. Empirical results on benchmark datasets demonstrate the superiority of our approach in both language modeling and auto-regressive generation. While the proposed method focuses on the fundamental text modeling, it is promising to extend to more applications such as dialogue generation, summarization and machine translation. It is also appealing to verify the approach on more recent network architectures such as Transformers.

REFERENCES

[1] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," in *Proceedings of the International Conference on Learning Representations*, 2013.

[2] J. Chung, K. Kastner, L. Dinh, K. Goel, A. C. Courville, and Y. Bengio, "A recurrent latent variable model for sequential data," in *Advances in neural information processing systems*, 2015, pp. 2980–2988.

[3] B. Zhang, D. Xiong, J. Su, H. Duan, and M. Zhang, "Variational neural machine translation," Preprint arXiv:1605.07869, 2016.

[4] J. Su, S. Wu, D. Xiong, Y. Lu, X. Han, and B. Zhang, "Variational recurrent neural machine translation," in *Procddings of the 34-th AAAI Conference on Artificial Intelligence*, 2018.

[5] P. Z. Wang and W. Y. Wang, "Neural gaussian copula for variational autoencoder," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, 2019, pp. 4332–4342.

[6] B. Li, J. He, G. Neubig, T. Berg-Kirkpatrick, and Y. Yang, "A surprisingly effective fix for deep latent variable modeling of text," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, 2019.

[7] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[8] H. Fu, C. Li, X. Liu, J. Gao, A. Çelikyilmaz, and L. Carin, "Cyclical annealing schedule: A simple approach to mitigating kl vanishing," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, 2019, pp. 240–250.

[9] J. He, D. Spokoyny, G. Neubig, and T. Berg-Kirkpatrick, "Lagging inference networks and posterior collapse in variational autoencoders," Preprint arXiv:1811.00135, 2019.

[10] H. Bahuleyan, L. Mou, O. Vechtomova, and P. Poupart, "Variational attention for sequence-to-sequence models," in *Proceedings of the 27th International Conference on Computational Linguistics*, 2018, pp. 1672–1682.

[11] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *Proceedings of the 3rd International Conference on Learning Representations*, 2015.

[12] T. Luong, H. Pham, and C. D. Manning, "Effective approaches to attention-based neural machine translation," in *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2015.

[13] S. R. Bowman, L. Vilnis, O. Vinyals, A. M. Dai, R. Józefowicz, and S. Bengio, "Generating sentences from a continuous space," in *Proceedings of the 20th SIGNLL Conference on Computational Natural Language Learning*, 2015, pp. 10–21.

[14] D. P. Kingma, T. Salimans, and M. Welling, "Improved variational inference with inverse autoregressive flow," no. Advances in the 30-th Neural Information Processing Systems, 2017.

[15] Z. Yang, Z. Hu, R. Salakhutdinov, and T. Berg-Kirkpatrick, "Improved variational autoencoders for text modeling using dilated convolutions," in *Proceedings of the 34th International Conference on Machine Learning*, 2017, pp. 3881–3890.

[16] S. Semeniuta, A. Severyn, and E. Barth, "A hybrid convolutional variational autoencoder for text generation," in *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*, 2017.

[17] J. Xu and G. Durrett, "Spherical latent spaces for stable variational autoencoders," in *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 2018, pp. 4503–4513.

[18] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, "Neural discrete representation learning," in *Advances in Neural Information Processing Systems*, 2017, pp. 6306–6315.

[19] A. Razavi, A. van den Oord, and O. Vinyals, "Generating diverse high-fidelity images with vq-vae-2," in *Advances in Neural Information Processing Systems*, 2019.

[20] A. van den Oord, N. Kalchbrenner, L. Espeholt, K. Kavukcuoglu, O. Vinyals, and A. Graves, "Conditional image generation with pixelcnn decoders," in *Advances in Neural Information Processing Systems*, 2016.

[21] A. Roy, A. Vaswani, A. Neelakantan, and N. Parmar, "Theory and experiments on vector quantized autoencoders," Preprint arXiv:1805.11063, 2018.

[22] Y. Bengio, N. Léonard, and A. Courville, "Estimating or propagating gradients through stochastic neurons for conditional computation," Preprint arXiv:1308.3432, 2013.

[23] M. P. Marcus, B. Santorini, and M. A. Marcinkiewicz, "Building a large annotated corpus of english: The penn treebank," *Computational Linguistics*, 1993.

[24] S. R. Bowman, G. Angeli, C. Potts, and C. D. Manning, "A large annotated corpus for learning natural language inference," *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, 2015.

[25] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," *OpenAI Blog*, vol. 1, no. 8, p. 9, 2019.

[26] J. Li, M. Galley, C. Brockett, J. Gao, and B. Dolan, "A diversity-promoting objective function for neural conversation models," Preprint arXiv:1510.03055, 2015.

[27] H. Kim, A. Mnih, J. Schwarz, M. Garnelo, A. Eslami, D. Rosenbaum, O. Vinyals, and Y. W. Teh, "Attentive neural processes," in *Proceedings of the International Conference on Learning Representations*, 2019.

[28] Y. Deng, Y. Kim, J. Chiu, D. Guo, and A. Rush, "Latent alignment and variational attention," in *Advances in Neural Information Processing Systems*, 2018.

[29] N. Subramani, S. Bowman, and K. Cho, "Can unconditional language models recover arbitrary sentences?" in *Advances in Neural Information Processing Systems*, 2019, pp. 15 258–15 268.

[30] J. T. Rolfe, "Discrete variational autoencoders," in *Proceedings of the International Conference on Learning Representations*, 2017.

[31] A. Vahdat, W. G. Macready, Z. Bian, and A. Khoshaman, "Dvae++: Discrete variational autoencoders with overlapping transformations," in *Proceedings of the International Conference on Machine Learning*, 2018.

[32] A. Vahdat, E. Andriyash, and W. G. Macready, "Dvae#: Discrete variational autoencoders with relaxed boltzmann priors," in *Advances in Neural Information Processing Systems*, 2018.