POSTERIOR VARIANCE-PARAMETERISED GAUSSIAN DROPOUT: IMPROVING DISENTANGLED SEQUENTIAL AUTOENCODERS FOR ZERO-SHOT VOICE CONVERSION

Yin-Jyun Luo, Simon Dixon

Centre for Digital Music, Queen Mary University of London

ABSTRACT

The class of disentangled sequential auto-encoders factorises speech into time-invariant (global) and time-variant (local) representations for speaker identity and linguistic content, respectively. Many of the existing models employ this assumption to tackle zero-shot voice conversion (VC), which converts speaker characteristics of any given utterance to any novel speakers while preserving the linguistic content. However, balancing capacity between the two representations is intricate, as the global representation tends to collapse due to its lower information capacity along the time axis than that of the local representation. We propose a simple and effective dropout technique that applies an information bottleneck to the local representation via multiplicative Gaussian noise, in order to encourage the usage of the global one. We endow existing zero-shot VC models with the proposed method and show significant improvements in speaker conversion in terms of speaker verification acceptance rate and comparable or better intelligibility measured in character error rate.

***Index Terms*— ** Disentanglement, variational autoencoder, posterior collapse, zero-shot voice conversion, Gaussian dropout

1. INTRODUCTION

Capturing semantically meaningful features of data is of great interest in representation learning [1]. In speech, the vocal characteristic of speakers can be represented as a time-invariant global attribute that is used to condition speech generation [2, 3, 4, 5, 6, 7].

For zero-shot voice conversion (VC), the ability to disentangle global speaker attributes from spoken content has delivered promising results [8, 9, 10, 11, 12, 13]. The objective of VC is to convert the speaker characteristic of a source utterance to that of a target utterance and preserve the phonetic or linguistic content of the source utterance, so that the converted speech sounds as if the target speaker spoke the source utterance. The zero-shot scenario is a challenging setup that requires a system to generalise to source and target speakers that are not included in training datasets. Capturing the speaker and linguistic information in separate representations, disentangling the two semantics, facilitates the application, whereby the conversion is as simple as preserving the linguistic representation derived from the source and replacing the global inferred from the target.

A key assumption underlying existing approaches is that while the speaker is represented as a global feature vector, the time-varying linguistic content is represented as a sequence of frame-level representations. Although the discrepancy in temporal resolution encourages allocating the global and local information separately, it also causes a capacity gap, which results in challenges in practice. In particular, the sequence that is supposed to exclusively capture the local linguistic information can learn the global speaker information. In extreme cases, the local representation can take over all the necessary information for speech modelling and render the global representation useless [12, 13, 14].

A common strategy to tackle the issue is to leverage a speaker encoder that is pre-trained with a supervised speaker discriminative task [4, 15], so that the speaker embedding carries sufficient information [8, 16]. To avoid relying on a model that is pre-trained using large annotated datasets, some approaches task the speaker encoder with a contrastive objective, which pulls closer the embeddings of different utterances spoken by the same speaker and repels those by different speakers [17, 18]. Various approaches have been proposed to bottleneck the local representation to prevent leakage of speaker information, including tuning dimensionality [8], applying instance normalisation (IN) [9], learning a vector-quantised (VQ) representation [11], or combining strategies [10].

Disentangled sequential autoencoders (DSAEs) [19, 20, 21, 14] are built on the key assumption of global-local separation. Unlike the instances mentioned above, DSAEs are principled probabilistic models that equip both latent variables with prior distributions, optimised using the variational autoencoder (VAE) [22]. However, the prior distribution poses an additional bottleneck to the global or speaker representation, which can more easily lead to collapse, and careful adjustment of prior regularisation strength is necessary to train the DSAE for zero-shot VC [12, 13]. Despite the tuning effort, the model remains fully unsupervised and shows impressive results without a pre-trained speaker encoder nor a VQ module. Similarly to the transition from deterministic AEs to VAEs in representation learning [23], an explicit prior distribution over the latent space of the speaker is favourable, as it allows for unconditional speaker sampling and, presumably, a smoother interpolation between the characteristics of the speakers [6].

In this paper, we introduce posterior variance-parameterised Gaussian dropout, pvpGD, to counteract the collapse of the global representation. [1] In particular, we endow existing zero-shot VC models with a prior distribution over the global latent space, turning the speaker representation into a stochastic latent variable as in the DSAE, and apply pvpGD to the local linguistic representation. Unlike conventional GD which requires tuning of the dropout strength [24, 25, 26], pvpGD is parameterised by the variance of the posterior distribution over the global latent variable. In principle, it is applicable as long as the posterior of the global latent follows a Gaussian distribution. We apply pvpGD to DSAEs including their speaker-supervised, IN, and VQ counterparts across a wide range of prior regularisation strengths and show significant improvements in terms of speaker conversion rate with comparable or better linguistic information preservation measured in character error rate.

The first author is a research student at the UKRI CDT in AI and Music, and would like to thank Emmanouil Benetos and Chin-Yun Yu for their valuable feedback on improving the initial manuscript.

[1] Code available at https://github.com/yjlolo/DSAE-VC.

2. RELATED WORK

2.1. Disentangled sequential autoencoder

The DSAE assumes that an observed time sequence is sampled from a global and a local latent variable [19]. Given a sequence $\mathbf{x}_{1:T}$ of length T, a sequence-level vector $\mathbf{s}$ describes global factors of variation, while a sequence of frame-level latent variables $\mathbf{z}_{1:T}$ associates with dynamic and local features. Given the model parameters θ, the joint distribution $p_\theta(\mathbf{x}_{1:T}, \mathbf{z}_{1:T}, \mathbf{s})$ is written as $\prod_{t=1}^{T} p_\theta(\mathbf{x}_t|\mathbf{s}, \mathbf{z}_t) p_\theta(\mathbf{z}_t|\mathbf{z}_{<t}) p_\theta(\mathbf{s})$ where $\mathbf{z}_t$ and $\mathbf{s}$ are marginally independent to encourage disentanglement. To learn the model, we can use the VAE framework [22] and maximise:

$$\begin{aligned}\mathcal{L}(\theta, \phi; \mathbf{x}_{1:T}) = &\sum_{t=1}^{T} \mathbb{E}_{q_\phi(\mathbf{z}_t|\mathbf{x}_{1:T}) q_\phi(\mathbf{s}|\mathbf{x}_{1:T})} \big[\log p_\theta(\mathbf{x}_t|\mathbf{z}_t, \mathbf{s}) \big] \\ &- \sum_{t=1}^{T} \mathbb{E}_{q_\phi(\mathbf{z}_t|\mathbf{x}_{1:T})} \big[\mathcal{D}_{\mathrm{KL}}\big(q_\phi(\mathbf{z}_t|\mathbf{x}_{1:T}) \| p(\mathbf{z}_t) \big) \big] \\ &- \beta \mathcal{D}_{\mathrm{KL}}\big(q_\phi(\mathbf{s}|\mathbf{x}_{1:T}) \| p(\mathbf{s}) \big),\end{aligned} \tag{1}$$

where the dynamic prior is simplified as $p_\theta(\mathbf{z}_{1:T}) = \prod_{t=1}^{T} p(\mathbf{z}_t)$. For simplicity, we can set the prior distributions $p(\mathbf{z}_t)$ and $p(\mathbf{s})$ as $\mathcal{N}(\mathbf{0}, \mathbf{I})$. The posterior distribution $q_\phi(\mathbf{z}_{1:T}, \mathbf{s}|\mathbf{x}_{1:T})$ is factorised as $q_\phi(\mathbf{s}|\mathbf{x}_{1:T}) \prod_{t=1}^{T} q_\phi(\mathbf{z}_t|\mathbf{x}_{1:T})$. The likelihood $p_\theta(\mathbf{x}_t|\mathbf{z}_t, \mathbf{s})$ is a diagonal Gaussian parameterised by the decoder with parameters θ. The posteriors $q_\phi(\cdot|\mathbf{x}_{1:T})$ are also diagonal Gaussians whose parameters are outputs of encoders ϕ. Similarly to β-VAE [27], β is introduced to flexibly control the information capacity of $\mathbf{s}$. The model assumption of Eq. (1) has been applied to learn disentangled representations of sequences in various domains [20, 21, 14].

2.2. Zero-shot voice conversion

In zero-shot VC, speaker identity and linguistic content can be encoded by $q_\phi(\mathbf{s}|\mathbf{x}_{1:T})$ and $q_\phi(\mathbf{z}_{1:T}|\mathbf{x}_{1:T})$, respectively. Upon inference, the conversion is as simple as keeping the linguistic representation $\mathbf{z}_{1:T}$ derived from a source utterance and replacing the speaker representation $\mathbf{s}$ of a target utterance. In practice, optimising Eq. (1) faces the challenge of information balance. By construction, $\mathbf{s}$ has a lower information capacity than $\mathbf{z}_{1:T}$ due to the temporal resolution discrepancy. Therefore, the speaker information can be captured by $\mathbf{z}_{1:T}$, leading to inferior conversion results. While a small β mitigates the issue, it leaves $q_\phi(\mathbf{s}|\cdot)$ weakly constrained. On the other hand, a large β can result in $\mathcal{D}_{\mathrm{KL}}\big(q_\phi(\mathbf{s}|\cdot)\|p(\mathbf{s})\big) \to 0$, or the standard deviation of the posterior $q_\phi(\mathbf{s}|\cdot)$ approaching unity, that is, $\sigma_\phi(\mathbf{s}) \to \mathbf{1}$, rendering the speaker representation $\mathbf{s}$ useless.

Existing models adopt the DSAE framework with modifications in order to retain speaker information in the global latent space. AutoVC [8] drops both terms of $\mathcal{D}_{\mathrm{KL}}\big(\cdot \| \cdot\big)$ or Kullback–Leibler divergence (KLD) from Eq. (1) and alternatively constrains the capacity of $\mathbf{z}_{1:T}$ by limiting both temporal and spatial dimensionality, and replaces the speaker encoder $q_\phi(\mathbf{s}|\cdot)$ with a pre-trained model on a discriminative task with respect to speaker. Other forms of speaker supervision are also considered to learn $\mathbf{s}$ through auxiliary or adversarial objectives [17, 28]. AdaIN-VC [9] applies the IN layer [29] to $\mathbf{z}_{1:T}$ to normalise global statistics and sets $\beta = 0$ to remove the bottleneck in the speaker representation. VQ is applied to $\mathbf{z}_{1:T}$ which replaces the second term in Eq. (1) with the discretisation bottleneck [30], which imposes a stronger constraint to avoid capturing the speaker information [10, 11].

Although these models differ in the availability of speaker information, the common strategy is to constrain $\mathbf{z}_{1:T}$ and promote the use of $\mathbf{s}$. On the other hand, there is recent work that investigates the capability of the original DSAE for zero-shot VC by optimising Eq. (1), preserving the prior distributions for both latent variables, and remaining free of speaker supervision and auxiliary loss functions [12, 13]. The main advantages are its simplicity and the fact that it is fully unsupervised. The preservation of $p(\mathbf{s})$ also allows for unconditional sampling of new speakers and a smoother interpolation between speaker characteristics [3, 6]. However, as discussed previously, it is not trivial to strike a capacity balance via β [12, 13].

2.3. Gaussian dropout as an information bottleneck

A desideratum of representation is being expressive about the tasks of interest while being maximally compressive about the data [1, 31]. Existing work has explored the application of Gaussian noise to representation as a bottleneck for this purpose [25, 32]. Inspired by the Gaussian dropout (GD) layer that applies multiplicative Gaussian noise to the representation [25], we seek a simple and effective method to constrain $\mathbf{z}_{1:T}$ and promote the use of $\mathbf{s}$.

3. METHOD

We develop a novel parameterisation of the variance of the multiplicative Gaussian noise, which translates to the strength of dropout or bottleneck and avoids additional hyperparameters.

We denote the standard deviation of the diagonal Gaussian $q_\phi(\mathbf{s}|\cdot)$ as $\sigma_s := \sigma_\phi(\mathbf{s}) = (\sigma_s^{(1)}, \ldots, \sigma_s^{(D_s)}) \in \mathbb{R}^{D_s}$, where D_s is the size of the global latent space. It follows:

$$\log \sigma_{pvpGD} := \frac{1}{D_s} \sum_{d=1}^{D_s} \log \sigma_s^{(d)} = \log \big[\prod_{d=1}^{D_s} \sigma_s^{(d)} \big]^{\frac{1}{D_s}}. \tag{2}$$

That is, we propose pvpGD whose standard deviation is posterior variance-parameterised, or simply the geometric mean of σ_s.

We can apply pvpGD to any variable, and we are interested in constraining $\mathbf{z}_t$ for our application. We apply multiplicative Gaussian noise to $\mathbf{z}_t \sim q_\phi(\mathbf{z}_t|\cdot)$ and obtain a perturbed local latent as:

$$\mathbf{z}'_t = \mathbf{z}_t \times \epsilon, \text{where } \epsilon \sim \mathcal{N}\big(\mathbf{1}, \sigma^2_{pvpGD}\mathbf{I}\big). \tag{3}$$

Unlike conventional GD [25, 26], pvpGD does not incur additional hyperparameters, such as dropout probability. Eq. (2) suggests that calculating σ^2_{pvpGD} is as trivial as exponentiating the arithmetic mean of $\log \sigma_s^2$, and $\log \sigma_s^2$ is usually an output of a fully connected layer that parameterises $q_\phi(\mathbf{s}|\cdot)$ in common VAE implementations.

We provide an idea of how pvpGD mitigates posterior collapse. Over-regularisation driving $\mathcal{D}_{\mathrm{KL}}\big(q_\phi(\mathbf{s}|\cdot)\|p(\mathbf{s})\big) \to 0$ as mentioned in Section 2.2 is mitigated as a result of σ_{pvpGD}. This is because increasing σ_s results in a stronger bottleneck applied to $\mathbf{z}_t$ according to Eq. (2) and Eq. (3). Since collapsing $\mathbf{s}$ also sabotages the capacity of $\mathbf{z}_t$, the model is more likely to limit the expansion of σ_s, thus mitigating collapse. Meanwhile, making the noise σ_{pvpGD} infinitely small to gain capacity for $\mathbf{z}_t$, effectively removing the stochasticity of $q_\phi(\mathbf{s}|\cdot)$, is penalised by $\mathcal{D}_{\mathrm{KL}}\big(q_\phi(\mathbf{s}|\mathbf{x}_{1:T})\|p(\mathbf{s})\big)$ from Eq. (1) through the standard Normal $p(\mathbf{s})$.

In other words, the proposed pvpGD counteracts the collapse of $\mathbf{s}$ by imposing a bottleneck on $\mathbf{z}_t$, and the strength of the bottleneck is parameterised by the capacity of $\mathbf{s}$. By coupling the capacity of the two latent variables, the model is forced to strike a balance in which both the expansion and the shrinkage of σ_s are limited, so that $\mathbf{s}$ is constrained, but at the same time carries a suitable amount of information. This is empirically verified in Fig. 3.

4. EXPERIMENTS

We apply pvpGD to AdaIN-VC [9] and DSAE [12, 13] as well as its speaker-supervised and VQ counterparts. In zero-shot VC, we provide empirical evidence that pvpGD mitigates the collapse of speaker representation across a range of β.

4.1. Datasets

We use VCTK [33] and a split of 99 and 10 English speakers in the training and testing sets, respectively. 10% of the training data is used for validation. There are equal numbers of male and female speakers in the test set. In addition to reserving speakers for testing, we also randomly excluded 47 of the total of 500 unique utterances from the training set.

In summary, we have $32,822$ and $4,011$ utterances in the training and testing sets, respectively. All testing data has unseen speakers, 378 utterances of which have spoken content not included in the training set. We also include the 25 utterances that were used for intra-lingual conversion in VCC2020 [34] only for evaluation.

We downsample recordings to $22,050$Hz and extract the logarithmic mel spectrogram of 80 filter banks from a short-time Fourier transform with a window length of 1024 and a hop size of 256.

4.2. Evaluation

We construct $4,011$ random source-target pairs from the VCTK test set. All speakers are unseen, and 378 pairs are conversions between unseen spoken content. VCC2020 provides another 400 source-target pairs for inter-dataset evaluation.

In zero-shot VC, we report the speaker acceptance rate (SAR) to measure the success of speaker conversion and the character error rate (CER) to gauge the preservation of spoken content [16, 35]. SAR is the ratio of converted utterances that are accepted by a speaker verification system (SV). We build separate SVs for the VCTK and VCC2020 test sets, using speaker embeddings derived from an existing pre-trained speaker encoder. [2] For CER, we use a pre-trained wav2vec 2.0. [3]

4.3. Implementation of common modules

We follow the AdaIN-VC speaker encoder [9] to implement $q_\phi(\mathbf{s}|\cdot)$. First, the input mel spectrogram is run through a convolution bank that concatenates the output of eight modules of one-dimensional convolution (Conv). The concatenated output is then processed by six blocks of Conv and subsampling layers followed by a temporal average pooling to obtain an utterance-level embedding. Finally, the global embedding is projected onto a space of size $D_s = 256$, which is then fed to two FC layers that make up the Gaussian layer and output μ_s and σ_s, respectively, to parameterise $q_\phi(\mathbf{s}|\cdot)$.

Adapted from a VQ model for acoustic unit discovery [36], the local encoder $q_\phi(\mathbf{z}_t|\cdot)$ consists of a Conv with a stride factor of two, which reduces the length of the resulting $\mathbf{z}$ to half of T. The Conv layer is followed by five blocks of layer normalisation, ReLU activation, and an FC layer. The final FC layer projects the intermediate embedding $\mathbf{e}_t$ in the local latent space of size $D_z = 64$, which is then fed to a Gaussian layer to output μ_z and σ_z at each time step.

The decoder $p_\theta(\mathbf{x}_t|\mathbf{z}_t, \mathbf{s})$ follows the original AutoVC [8], which first recovers the temporal resolution by upsampling $\mathbf{z}_t$ (or perturbed $\mathbf{z}'_t$) by a factor of two based on linear interpolation and repeats $\mathbf{s}$ for T steps. A 512-dimensional LSTM takes as input the concatenation of the two representations, followed by three blocks of Conv and batch normalisation (BN). Two layers of 1024-dimensional LSTM and an FC layer finally project the embedding back to the size of the mel spectrogram $D_x = 80$ followed by a Gaussian layer that outputs μ_x, while σ_x is fixed in practice. A PostNet consisting of five blocks of Conv and BN refines the reconstruction μ_x by predicting its difference from the input mel spectrogram. We use the pre-trained MelGAN [4] to invert the reconstructed mel spectrogram back to the audio waveform.

4.4. Implementation of individual models

AdaIN-VC extracts $\mathbf{s}$ from the described speaker encoder. We optimise the model using Eq. (1), which is a principled probabilistic extension of the original model [9] recovered when $\beta = 0$. The content encoder precedes the Gaussian layer with an IN. Instead of having the decoder take as input the concatenation of $\mathbf{z}$ and $\mathbf{s}$, AdaIN-VC retains speaker information by inserting adaptive IN layers into each of the three Conv blocks, which replaces the BN. Each adaptive IN layer performs an affine transformation with learnt parameters conditioned on $\mathbf{s}$ [9]. The application of pvpGD follows Eq. (2).

The DSAE is optimised by Eq. (1), configured with the speaker encoder, the content encoder, and the decoder that are described in Section 4.3. Again, Eq. (2) applies when pvpGD is used.

We also implement a speaker-supervised DSAE, or DSAE-S, as a reference. We replace the speaker encoder with the pre-trained model that is the same SV system in Section 4.2, similar to AutoVC [8]. In particular, we consider an alternative that removes the normalisation to the output of the pre-trained encoder and appends a Gaussian layer to fine-tune the speaker embedding. In this way, we preserve the principled probabilistic formulation and optimise the model with Eq. (1), which also allows the application of pvpGD.

Lastly, DSAE-VQ is the VQ variant that removes the Gaussian layer from the local encoder and learns a quantised $\mathbf{z}_t$ given the continuous $\mathbf{e}_t$ mentioned in Section 4.3 using a codebook of size 512. The second term of Eq. (1) is replaced by $\alpha \sum_{t=1}^{T} \|\mathbf{e}_t - \mathrm{sg}(\mathbf{z}_t)\|_2^2$ where $\alpha = 0.25$ and $\mathrm{sg}(\cdot)$ denotes the stop gradient, following existing work [36, 37]. When activated, pvpGD is applied to $\mathbf{e}_t$ instead of $\mathbf{z}_t$ to avoid additional training instability on top of quantisation.

4.5. Optimisation

We employ Adam [38] with a batch size of 256. Each data point is a segment of the mel spectrogram of length $T = 128$ or approximately one and a half seconds. We set the learning rate at 5e-4, $(\beta_1, \beta_2) = (0.9, 0.999)$, and a gradient clipping value of three. [5] The training is terminated if Eq. (1) evaluated using the validation set stops improving for a patience of 13k steps.

5. RESULTS

5.1. Applying pvpGD to different models and values of β

We present the main result in Fig. 1. Each model is evaluated with three random seeds. The model with an unconstrained speaker embedding ($\beta = 0$) is also included, where the proposed pvpGD is not applicable due to the fact that Gaussian noise is parameterised by the variance of a stochastic speaker latent variable. [6]

[2] https://github.com/resemble-ai/Resemblyzer

[3] https://huggingface.co/facebook/wav2vec2-large-960h-lv60-self

[4] https://github.com/descriptinc/melgan-neurips/tree/master

[5] Settings from https://github.com/KimythAnly/AGAIN-VC [39].

[6] Audio samples are provided at https://shorturl.at/chpuy.

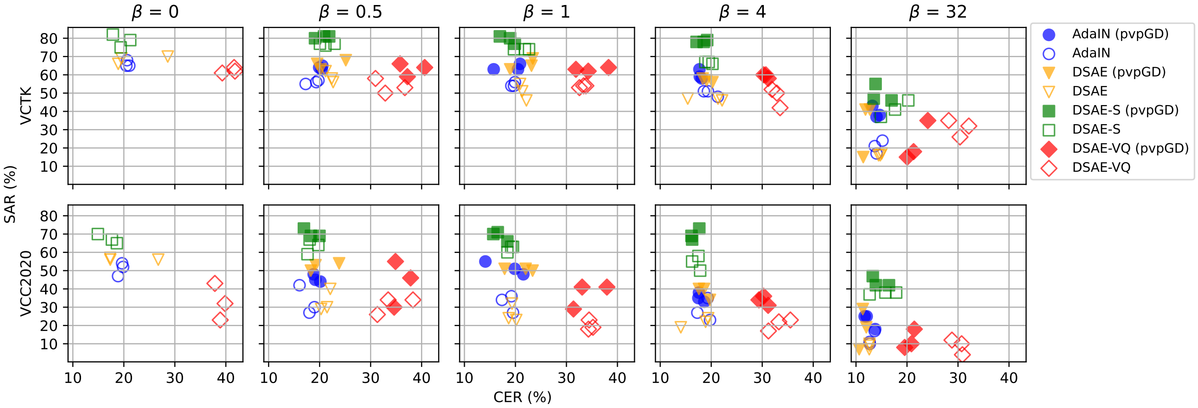

Fig. 1. Evaluation of zero-shot VC on VCTK and VCC2020 using three random seeds per model. Speaker conversion in terms of speaker acceptance rate (SAR, higher is better) versus preservation of linguistic content in terms of character error rate (CER, lower is better) across β.

When $\beta > 0$, models equipped with pvpGD outperform their vanilla counterparts in terms of SAR, that is, most of the filled marks are placed north of the unfilled marks within each shape group. Exceptions are one run of DSAE and DSAE-VQ at $\beta = 32$. In terms of CER, we observe no systemic difference between filled and unfilled marks within each shape group. In fact, pvpGD improves CER in some cases, such as DSAE at $\beta = 32$, DSAE-S at $\beta = 1$, and DSAE-VQ at $\beta \in \{4, 32\}$.

Overall, the speaker-supervised model DSAE-S is considered as a reference that outperforms unsupervised models, and pvpGD is able to narrow the gap between them. Meanwhile, DSAE-VQ attains less intelligible utterances due to the VQ module, i.e. the diamond markers are east of the rest. Lastly, the efficacy of pvpGD depends on a reasonable range of β, beyond which (e.g., $\beta = 32$) the advantage is less consistent. The result identifies the sensitivity of the model to β, aligning with previous work [12, 13], and shows that pvpGD mitigates the issue without compromising the CER.

Although models with $\beta = 0$ generally reach decent SAR with an unconstrained speaker representation, AdaIN-VC can underperform its pvpGD counterpart at $\beta = 1$ in terms of CER at a comparable SAR. Similar results can be observed for DSAE and DSAE-VQ. In addition to the favourable properties described in Section 1, $\beta > 0$ promotes disentangled features between dimensions of the speaker latent variable [6, 27], facilitating a variety of applications.

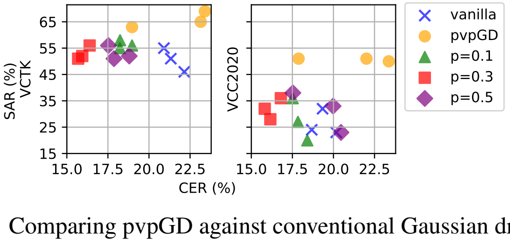

Fig. 2. Comparing pvpGD against conventional Gaussian dropout.

5.2. Comparing pvpGD with vanilla Gaussian dropout

In Fig. 2, we use DSAE with $\beta = 1$ and show that pvpGD outperforms conventional multiplicative GD [25] in terms of SAR at dropout strength $p = \{0.1, 0.3, 0.5\}$, where parameterisation $\sigma^2_{GD} = \frac{p}{1-p}$ replaces σ^2_{pvpGD} in Eq. (3). It shows that pvpGD trades off approximately a 5% loss of CER for a 10% gain of SAR that increases to 20% in VCC2020. One run of pvpGD does not bear the loss of intelligibility. Among vanilla GD, $p = 0.3$ suggests the best trade-off between SAR and CER. Conversely, pvpGD strikes a decent trade-off without introducing additional hyperparameters.

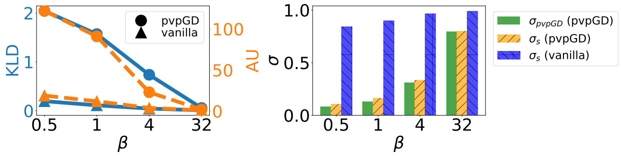

Fig. 3. Left: KLD (solid) and AU (dashed). Right: σ_s and σ_{pvpGD}.

5.3. Limiting expansion of posterior variance

Last but not least, we verify our intuition of pvpGD explained in Section 3 through Fig. 3 using DSAE. We report the average KLD $\mathbb{E}_{\mathbf{x}}\big[\mathcal{D}_{\mathrm{KL}}\big(q_\phi(\mathbf{s}|\mathbf{x})\|p(\mathbf{s})\big)\big]$ and the active unit (AU) [40] in the left panel, and the arithmetic mean of σ_s as well as σ_{pvpGD} in the right panel. It shows that pvpGD increases the average KLD by preventing $\sigma_s \to 1$ as shown in the right panel. Meanwhile, AU measures the activity of each of the $D_s = 256$ dimensions in the speaker latent space as $A_s = \mathrm{Cov}_{\mathbf{x}}\big(\mathbb{E}_{s\sim q_\phi(s|\mathbf{x})}\big[s\big]\big)$, and a dimension is considered active if $A_s > 0.01$. As shown in the left panel, AU trends similarly to KLD, and pvpGD maintains the active rate until $\beta = 32$ where the gap is closed, which aligns with the SAR in Fig. 1. However, note that the AU of pvpGD at $\beta = 32$ is two versus one for the vanilla counterpart, which is an increase in information capacity of 100%.

In summary, a reasonable magnitude of β is a prerequisite for pvpGD to deliver a consistent improvement. When the penalty of KLD caused by shrinkage of σ_s, exacerbated by upscaling β, outweighs the decrease in reconstruction loss, the advantage caused by pvpGD is less significant. Therefore, looking for ways that efficiently exploit the gained information capacity, as indicated in Fig. 3, and help to achieve a greater improvement in reconstruction, is a promising future direction for enhancing the efficacy of pvpGD.

6. CONCLUSION

We have proposed pvpGD, a simple and effective GD that mitigates posterior collapse of a latent variable. It does not introduce additional parameters and is applicable to latent-variable models in general. We have demonstrated its efficacy using zero-shot VC, and will study how to efficiently deploy pvpGD, e.g., selecting features to which it is applied or designing new architectures to further expand the range of β within which it can deliver consistent improvement.

7. REFERENCES

[1] Y. Bengio, A. Courville, and P. Vincent, "Representation Learning: A Review and New Perspectives," in *PAMI*, 2013.

[2] W.-N. Hsu, Y. Zhang, and J. Glass, "Unsupervised Learning of Disentangled and Interpretable Representations from Sequential Data," in *NeurIPS*, 2017.

[3] D. Stanton, M. Shannon, S. Mariooryad, R. J. Skerry-Ryan, E. Battenberg, T. Bagby, and D. Kao, "Speaker Generation," in *ICASSP*, 2022.

[4] Y. Jia, Y. Zhang, R. Weiss, Q. Wang, J. Shen, F. Ren, Z. Chen, P. Nguyen, R. Pang, I. Lopez Moreno, and Y. Wu, "Transfer Learning from Speaker Verification to Multispeaker Text-To-Speech Synthesis," in *NeurIPS*, 2018.

[5] Y. Chen, Y. Assael, B. Shillingford, D. Budden, S. Reed, H. Zen, Q. Wang, L. C. Cobo, A. Trask, B. Laurie, C. Gulcehre, A. van den Oord, O. Vinyals, and N. Freitas, "Sample Efficient Adaptive Text-to-Speech," in *ICLR*, 2019.

[6] W.-N. Hsu, Y. Zhang, R. J. Weiss, H. Zen, Y. Wu, Y. Wang, Y. Cao, Y. Jia, Z. Chen, J. Shen, P. Nguyen, and R. Pang, "Hierarchical Generative Modeling for Controllable Speech Synthesis," in *ICLR*, 2019.

[7] M. Ravanelli and Y. Bengio, "Learning Speaker Representations with Mutual Information," in *Interspeech*, 2019.

[8] K. Qian, Y. Zhang, S. Chang, X. Yang, and M. Hasegawa-Johnson, "AutoVC: Zero-Shot Voice Style Transfer with Only Autoencoder Loss," in *ICML*, 2019.

[9] J.-C. Chou and H.-Y. Lee, "One-Shot Voice Conversion by Separating Speaker and Content Representations with Instance Normalization," in *Interspeech*, 2019, pp. 664–668.

[10] D.-Y. Wu and H.-Y. Lee, "One-Shot Voice Conversion by Vector Quantization," in *ICASSP*, 2020.

[11] A. Tjandra, R. Pang, Y. Zhang, and S. Karita, "Unsupervised Learning of Disentangled Speech Content and Style Representation," in *Interspeech*, 2021.

[12] J. Lian, C. Zhang, and D. Yu, "Robust Disentangled Variational Speech Representation Learning for Zero-Shot Voice Conversion," in *ICASSP*, 2022.

[13] H. Lu, D. Wang, X. Wu, Z. Wu, X. Liu, and H. Meng, "Disentangled Speech Representation Learning for One-Shot Cross-Lingual Voice Conversion Using β-VAE," in *SLT*, 2023.

[14] Y.-J. Luo, S. Ewert, and S. Dixon, "Towards Robust Unsupervised Disentanglement of Sequential Data — A Case Study Using Music Audio," in *IJCAI*, 2022.

[15] E. Variani, X. Lei, E. McDermott, I. L. Moreno, and J. Gonzalez-Dominguez, "Deep Neural Networks for Small Footprint Text-Dependent Speaker Verification," in *ICASSP*, 2014.

[16] W.-C. Huang, S.-W. Yang, T. Hayashi, and T. Toda, "A Comparative Study of Self-supervised Speech Representation Based Voice Conversion," in *JSTSP*, 2022.

[17] M. Luong and V. A. Tran, "Many-to-Many Voice Conversion Based Feature Disentanglement Using Variational Autoencoder," in *Interspeech*, 2021.

[18] S. Yuan, P. Cheng, R. Zhang, W. Hao, Z. Gan, and L. Carin, "Improving Zero-Shot Voice Style Transfer via Disentangled Representation Learning," in *ICLR*, 2020.

[19] Y. Li and S. Mandt, "Disentangled Sequential Autoencoder," in *ICML*, 2018.

[20] J. Bai, W. Wang, and C. P. Gomes, "Contrastively Disentangled Sequential Variational Autoencoder," in *NeurIPS*, 2021.

[21] S. Khurana, S. R. Joty, A. Ali, and J. Glass, "A Factorial Deep Markov Model for Unsupervised Disentangled Representation Learning from Speech," in *ICASSP*, 2019.

[22] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," in *ICLR*, 2014.

[23] M. Tschannen, O. Bachem, and M. Lucic, "Recent Advances in Autoencoder-Based Representation Learning," in *NeurIPS Workshop on Bayesian Deep Learning*, 2018.

[24] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: A Simple Way to Prevent Neural Networks from Overfitting," in *JMLR*, 2014.

[25] M. Rey and A. Mnih, "Gaussian Dropout as an Information Bottleneck Layer," in *NeurIPS Workshop on Bayesian Deep Learning*, 2021.

[26] S. Wang and C. Manning, "Fast Dropout Training," in *ICML*, 2013.

[27] I. Higgins, L. Matthey, A. Pal, C. Burgess, X. Glorot, M. Botvinick, S. Mohamed, and A. Lerchner, "β-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework," in *ICLR*, 2017.

[28] B. Nguyen and F. Cardinaux, "NVC-Net: End-To-End Adversarial Voice Conversion," in *ICASSP*, 2022.

[29] X. Huang and S. Belongie, "Arbitrary Style Transfer in Real-Time with Adaptive Instance Normalization," in *ICCV*, 2017.

[30] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, "Neural Discrete Representation Learning," in *NeurIPS*, 2017.

[31] N. Tishby, C. Pereira, and W. Bialek, "The Information Bottleneck Method," in *Allerton Conference on Communication, Control and Computation*, 2001.

[32] A. A. Alemi, I. Fischer, J. V. Dillon, and K. Murphy, "Deep Variational Information Bottleneck," in *ICLR*, 2017.

[33] C. Veaux, J. Yamagishi, and S. King, "The Voice Bank Corpus: Design, Collection and Data Analysis of a Large Regional Accent Speech Database," in *CASLRE*, 2013.

[34] Y. Zhao, W.-C. Huang, X. Tian, J. Yamagishi, R. K. Das, T. Kinnunen, Z. Ling, and T. Toda, "Voice Conversion Challenge 2020," in *ISCA Joint Workshop for the Blizzard Challenge and Voice Conversion Challenge 2020*, 2020.

[35] T.-H. Huang, J.-H. Lin, and H.-Y. Lee, "How Far Are We from Robust Voice Conversion: A Survey," in *SLT*, 2021.

[36] B. V. Niekerk, L. Nortje, and H. Kamper, "Vector-Quantized Neural Networks for Acoustic Unit Discovery in the ZeroSpeech 2020 Challenge," in *Interspeech*, 2020.

[37] D. Wang, L. Deng, Y. T. Yeung, X. Chen, X. Liu, and H. Meng, "VQMIVC: Vector Quantization and Mutual Information-Based Unsupervised Speech Representation Disentanglement for One-Shot Voice Conversion," in *Interspeech*, 2021.

[38] S. J. Reddi, S. Kale, and S. Kumar, "On the Convergence of Adam and Beyond," in *ICLR*, 2018.

[39] Y.-H. Chen, D.-Y. Wu, T.-H. Wu, and H.-Y. Lee, "Again-VC: A One-Shot Voice Conversion Using Activation Guidance and Adaptive Instance Normalization," in *ICASSP*, 2021.

[40] Y. Burda, R. B. Grosse Grosse, and R. Salakhutdinov, "Importance Weighted Autoencoders," in *ICLR*, 2016.