End-to-End Voice Conversion using Discrete Latent Representations: Final Report

G086 (s1873447, s1308389, s1837038)

Abstract

Voice Conversion (VC) is widely desirable across many industries and applications, including speaker anonymisation, film dubbing, gaming, and voice restoration for people who have lost their ability to speak. In this work we compare standard VAE, VQ-VAE and Gumbel VAE models as approaches to end-to-end VC on the Voice Conversion Challenge 2016 dataset. We assess reconstruction and VC performance on both spectral frames as obtained from a WORLD vocoder and on the raw waveform data. Based on evaluations obtained from human raters, our results confirm that the VQ-VAE model is a powerful method for learning discrete latent representations, which outperforms both a standard VAE as well as the more simple way of representing a discrete latent using a Gumbel softmax. However, we find that the VQ-VAE model did not make use of the provided speaker embeddings to perform VC. Here, we provide interpretations and possible future research directions to explore.

1. Introduction

The human voice conveys a multitude of non- or paralinguistic information about a speaker's identity, such as their age, gender, where they come from, or even their physical appearance (Kreiman & Sidtis, 2011). Voice conversion (VC) is a speech processing field aiming to develop tools for transforming a person's (or speech-synthesis system's) speech to another target person's voice, without changing the linguistic content (Abe et al., 1990). Applications include customised text-to-speech systems, speech-to-speech translation, film dubbing, speaker anonymisation, voice avatars in games, or speech-generating devices for people with speech impairments that aim to recreate the voices of their owners. Such systems are either trained on *parallel* data, where source and target speakers utter the same sentences, or *non-parallel* data where the sentences differ. Traditionally, approaches to parallel VC use time-frame alignment to generate labelled training data, while non-parallel VC may be approached with clustering methods to produce pseudo-parallel data or by using a conversion function that was learned from parallel data (Lorenzo-Trueba et al., 2018). More recently, however, VC has been approached as an unsupervised learning problem, where generative models can be trained end-to-end and do not require parallel data or clustering of source and target speakers utterances to perform VC.

At the heart of most recent unsupervised learning algorithms there is an underlying generative model (Liu et al., 2017). Generative modelling has recently made substantial advances in various domains, particularly images (Karras et al., 2018), audio (van den Oord et al., 2016), and video (Vondrick et al., 2016). While there are a number of different approaches to generative modelling, variational autoencoders (VAE) (Kingma & Welling, 2013) provide the advantage of modelling the latent space explicitly, as well as offer a more stable training than implicit latent space models such as generative adversarial networks (Goodfellow et al., 2014).

VAEs are an example of prescribed latent variable models that specify a distribution over latent variables, where assumptions about the latent density shape and noise are necessary. VAEs are trained to learn a low-dimensional latent representation of the inputs through a bottle-neck layer, minimising the reconstruction loss of the input data. Conceptually, the VC pipeline can be split into two models, a speech encoder and speech generator, where both can be treated as density learning problems. The task of the speech encoder is to find a compact and accurate representation of the linguistic content, and the task of the speech synthesizer is to generate speech that closely resembles the voice of the target speaker. This task separation makes a VAE model an attractive choice for voice conversion.

The latent variables in a VAE are usually represented as continuous samples from a Gaussian posterior distribution. However, many phenomena, such as phonemes, are better described as discrete variables (van den Oord et al., 2017). Phonemes are commonly defined as an equivalence class of physical sounds that change the meaning of words in a given language (Fodor et al., 1974). For example, the phoneme /t/ is pronounced differently in the words *cat* and *top*, but this physical difference in sound does not differentiate between meanings in English, and hence can be grouped in one phoneme (Traxler & Gernsbacher, 2011). Segmenting a stream of speech is known as a hard problem, as the physical representation of speech resembles a continuous stream, which people perceive in the form of distinct sounds (Monaghan & Christiansen, 2010).

Following the latent variable interpretation as discrete phonemes, in this work we investigate the use of discrete latent variable models for voice conversion. Representing

such discrete latent variables in a neural network is not straight-forward since the gradient cannot be computed directly during backpropagation. Vector-Quantised variational autoencoders (VQ-VAE) (van den Oord et al., 2017) have been proposed as an extension to VAEs by replacing the Gaussian sampling bottleneck with a vector quantisation layer to obtain discrete latent variables. Evaluations show promising performance on image and video generation as well as voice conversion on raw waveform data (van den Oord et al., 2017). Another recently proposed method replaces the Gaussian sampling layer of a VAE with a Gumbel softmax bottleneck to obtain a latent distribution over discrete values (Jang et al., 2016; Maddison et al., 2016; Dupont, 2018). To the best of our knowledge, there are no studies that compare these two different approaches of modelling discrete latents on speech processing tasks.

In this work, we aim to investigate the benefits of using discrete latent variable models for modelling speech data for voice conversion. First, we compare the models' capability to reconstruct spectral frame features extracted from the waveform data using a WORLD vocoder (Morise et al., 2016), and evaluated the models' ability to convert voice styles using spectral frame features. In further experiments we analyse which methods are able to model raw waveform data that is known to be particularly difficult to model.

In the next section we describe the VCC 2016 dataset used for the evaluation of our models, spectral feature extraction using the WORLD vocoder, mu-law encoding of waveform data, and evaluation and visualisation methods employed in our experiments. In Section 3, we outline the VAE, VQ-VAE and the Gumbel VAE models, as well as the network architectures. Our experimental designs, results and their discussion are presented in Section 4. In Section 5 we relate our findings to existing research and provide possible future research directions. Finally, we provide a summary of our results in Section 6.

We note that our initial plan to investigate an adversarial loss objective was replaced with a more thorough analysis of discrete latent models by including another model in our work – the Gumbel VAE. Also, due to the computationally expensive models required for effectively modelling raw speech data, our project direction has shifted more towards using a spectral frame representation, which can be modelled with simpler architectures.

2. Task and data

2.1. Data

For the experimental evaluation of the models we chose the Voice Conversion Challenge 2016 (VCC) dataset (Toda et al., 2016) [1]. The dataset contains approximately 13 minutes of studio recorded speech 10 professional US English speakers (5 male, 5 female) sampled at 16 kHz (Toda et al., 2016). Note that even though the dataset contains parallel samples for all speakers, we do not make use of this in our work and train our models in an unaligned fashion. Each spoken sentence is given as a separate file in Waveform Audio File Format (WAVE) which allows it to be easily processed by available tools.

We run experiments on two input representations: raw waveform, and spectral frames extracted using WORLD vocoder. The details of preprocessing required to process such input data are explained in the rest of this section.

2.2. Mu-law encoding

Mu-law encoding is a popular technique used in telecommunications and speech processing that allows us to increase the signal-to-noise ratio in the input data. For processing raw waveform data, we have transformed samples using a mu-law companding transformation, and then quantise to 256 discrete values:

$$f(\mathbf{x}) = \text{sign}(\mathbf{x}) \frac{\ln(1 + \mu|\mathbf{x}|)}{\ln(1 + \mu)},$$

where $\mathbf{x}$ was scaled between -1 and 1, and $\mu = 255$. Such an encoding scheme has been shown to allow the reconstruction of speech signals that sounds very similar to the original (van den Oord et al., 2016).

2.3. WORLD Vocoder

A vocoder is an audio processing technique that extracts characteristic components of speech in such a way that the speech can be resynthesised. WORLD (Morise et al., 2016) is a recently developed vocoder that is capable of fast, high quality reconstructions. In contrast to training a network directly on the reconstruction of the raw audio waveform, the WORLD vocoder can be used in a preprocessing stage to extract specific features of the speech. Specifically, there are three speech features that the WORLD vocoder extracts that are referred to as the spectral envelope, the fundamental frequency (F0) and the aperiodic parameters. Each of these features are extracted for short consecutive time frames of the input sample. Internally, WORLD uses a method called CheapTrick (Morise, 2015) to compute the spectral envelope, which involves applying a smoothing function to the log of power spectrum of the signal. The spectral envelope can therefore be understood as containing information about the amplitudes of each of the different frequencies that make up the signal within the frame. For all of our experiments the size of the Fast Fourier Transform window used within CheapTrick was set to 1024. The fundamental frequency is defined as the lowest frequency of the periodic signal. Finally, the aperiodic parameters contain the information about the component of the signal that cannot be characterised by frequencies such as transient sounds or noise.

In our work the network is trained to reconstruct only the spectral envelopes with the fundamental frequencies and

[1] Note that we initially considered to use the VCTK dataset, but switched to VCC for comparability with the other published reports and due to the large computational requirements to process VCTK.

aperiodic parameters being simply copied from the original input in as has been done in previous work Hsu et al. (2016; 2017). Using this representation of the audio has the benefit of providing the network with features that are more important to speech and therefore more relevant in the reconstruction process. Once a reconstruction of the features is obtained by one of our models it can be directly used by the WORLD vocoder to create an approximate reconstruction of the original waveform.

2.4. Evaluation

For the evaluation, we assess both the quality of the reconstruction as well as the voice conversion performance. As previous research indicates, quantitative measures such as reconstruction loss or mel-cepstral coefficients do not necessarily align with human judgements (Hsu et al., 2017). For this reason we employ human evaluation for the reconstruction and voice conversion quality. A total of 12 volunteer participants were recruited from fellow students (with no particular focus on speech processing experience). Informed consent was obtained from all participants and the study was approved by the University of Edinburgh Ethics board (RT #3507). For the evaluation of speech intelligibility on reconstructed samples we selected one sample per model from all speakers at random and let participants rate the samples on a five-point scale, following recommendations by (Streijl et al., 2016). For VC evaluation we selected one source-target pair per model at random, for each of which we converted one source speaker's sample [2] to the target speaker and let participants rate the similarity between the two samples on a five point scale.

In addition to human evaluations for VC, we also construct a cluster map of original samples from the source and target speakers, as well as reconstructed and voice-converted samples. The visualisation was performed on features containing statistics of the Mel-frequency cepstral coefficients (MFCC) (Mohamed et al., 2012) using t-distributed stochastic neighbour embedding (t-SNE) in a similar way to previous work (van der Maaten & Hinton, 2008). Similarly to how the spectral envelopes are extracted by WORLD, these coefficients are computed from the power spectrum of a sliding window over the input (Mohamed et al., 2012).

3. Methodology

Here, to perform an unaligned corpora voice conversion task, we train a model to reconstruct a source speaker's utterance. The conversion to a target speaker's voice is done without explicitly training the model, but is produced as a by-product of passing the speaker id to the generator. This enables the encoder to filter out the part of the signal unrelated to the linguistic contents of the speech, replicating the approach outlined in (van den Oord et al., 2017). In this work, we assume that the spectral frames $\mathbf{x}$ of the recordings come from a true probability density, where the densities for the source and target speakers are p_s and p_t. The VC task is then to learn a conditional distribution q_t on $\mathbf{x}_s \sim p_s$ such that $q_t \approx p_t$. Moreover, reflecting on the traditional VC pipelines that use an intermediary phoneme representation, we assume that speech can be encoded as a discrete low-dimensional code, namely a vector of phonemes or sounds. In the next section we describe a VAE that uses a Gaussian posterior latent layer, and in the rest of this section we propose two end-to-end VC methods that take advantage of the discrete latent interpretation of phonemes.

[2] With the constraint of not including samples from the intelligibility evaluation.

3.1. VAE

Variational Autoencoders (VAEs) (Kingma & Welling, 2013) specify a prior distribution over the latent variables. The goal of a VAE is then to approximate the parameters of the posterior distribution $p(\mathbf{z}|\mathbf{x})$ given the data, which is in the same distribution family as the prior. The latent distribution in standard VAEs are typically diagonal-covariance Gaussian, where μ and σ^2 are parameterised by the encoder network. During training, random samples are taken from the posterior $\mathbf{z} \sim q(\mathbf{z}|\mathbf{x})$ and passed to the generator, which aims to reconstruct the original input $\mathbf{x}$. This latent sampling layer works as a regularisation to the model encouraging similar input samples to be represented with similar latent representations. Computing gradients through the sampling layer requires an application of the reparameterisation trick. The reparameterisation trick forms a deterministic function of an auxiliary noise distribution and the posterior parameters from the encoder network, thus making backpropagation through random variables possible. In our work, we will refer to the decoder as *generator* G, since it is tasked to synthesise realistic data samples.

3.2. VQ-VAE for learning discrete latents

In VC we prefer to filter out most of the speaker-related variance and noise, and focus only on the discrete linguistic content, i.e. phonemes, that preserves the most relevant information for the task. Generally, training models with discrete layers is difficult since the gradient cannot be computed. In order to take advantage of discrete speech interpretation, we use Vector-Quantised VAE (VQ-VAE) (van den Oord et al., 2017) where a discrete embedding layer (VQ) is used to replace the posterior sampling layer in VAE. The embedding layer contains a dictionary of K D-dimensional real-valued embedding vectors, $\mathbf{e}_k$, where $k = 1..K$ and K is the number of discrete values allowed by the dictionary. The VQ layer takes the output from the encoder $\mathbf{z}_e = E(\mathbf{x})$ and calculates the discrete latent value $z_q = k$ by a nearest neighbour look-up in the dictionary of embeddings, where k is the index of the embedding vector $\mathbf{e}_k$ that minimises the distance to $\mathbf{z}_e$:

$$z_q = k = \arg\min_i \|\mathbf{z}_e - \mathbf{e}_i\|_2 \tag{1}$$

The closest embedding vector $\mathbf{e}_k$ (the one that minimises the objective described above) is then passed down as an input $\mathbf{z}_g = \mathbf{e}_k = \mathrm{VQ}(\mathbf{z}_e)$ to the generator network. For simplicity, the example above assumed that the input to

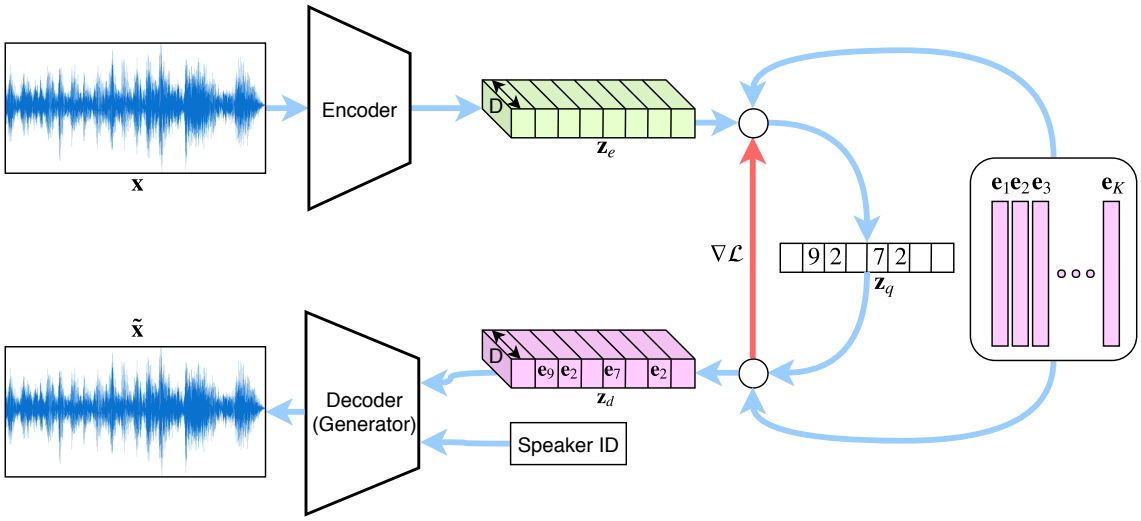

Figure 1. The VQ-VAE model architecture adapted for unaligned VC. The embedding dictionary on the far-right is used both when producing the discrete latents $\mathbf{z}_q$ and mapping them back to vectors $\mathbf{z}_d$ for an input to the generator. The generator receives a one-hot encoded speaker identity, therefore the encoder can filter out the speaker-dependent features and only pertain the linguistic content. The red arrow shows how the gradient bypasses the discrete representations $\mathbf{z}_q$ using a straight-through estimator.

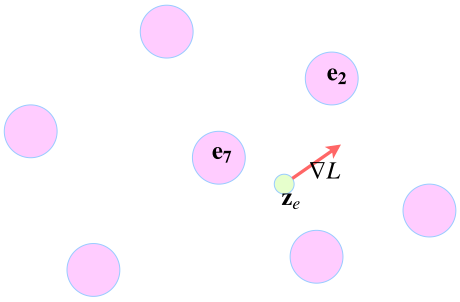

Figure 2. The figure shows how the straight-through gradient estimator can help the encoder to produce samples from a different category in the next forward pass. That is, if in the current run the vector $\mathbf{z}_e$ was assigned by nearest-neighbour distance to $\mathbf{e}_7$, then the straight-through gradient estimator can push the encoder output vector closer to $\mathbf{e}_2$ in the next forward run. This is a reasonable assumption, since the signs of the straight-through estimator are guaranteed to match those of the gradient (Bengio et al., 2013).

the VQ layer $\mathbf{z}_e$ was a D-dimensional vector mapped to a single latent variable z_q. However, in general the output can be any tensor whose channel is D-dimensional. Then, the nearest-neighbour quantisation described in Eq. 1 is applied to each element along the channel dimension. In the VC task it will be a $Z \times D$ tensor, where Z is the length of the down-sampled recording.

Since the gradient has to propagate through only a single discrete layer, VQ-VAE can use a straight-through gradient estimator copying the gradients from the generator input $\mathbf{z}_g$ to encoder output $\mathbf{z}_e$ bypassing the VQ layer. Bengio et al. (2013) show that the sign of the straight-through gradient estimator are guaranteed to be correct when propagating through a single layer and should contain useful information for training the outputs of the encoder as illustrated in the Figure 2.

The VQ-VAE loss function comprises of three terms – reconstruction loss, vector quantisation (embedding training) objective, and commitment loss:

$$\mathcal{L}_{\text{rec}} = \log(p(\mathbf{x}|E(\mathbf{x}))) \tag{2}$$

$$\mathcal{L}_{\text{VQ}} = \|\,\text{sg}[E(\mathbf{x})] - \mathbf{e}\|_2^2 \tag{3}$$

$$\mathcal{L}_{\text{com}} = \beta\|E(\mathbf{x}) - \text{sg}[\mathbf{e}]\|_2^2 \tag{4}$$

The reconstruction loss in Eq. 2 optimises the encoder and generator model parameters to maintain the contents of the original input, however due to the straight-through gradient estimation the embedding vectors $\mathbf{e}$ of the VQ layer are not updated. In order to train the embedding vectors, the vector quantisation objective in Eq. 3 is added to directly train the embedding vectors $\mathbf{e}$ to minimise the l_2 distance to the encoder output $\mathbf{z}_e$. Moreover, because the embedding space is dimensionless (it is not bound to any physical quantity) it can grow arbitrarily if the encoder does not train as fast as the embedding vectors. Therefore, the commitment loss in Eq. 4 is added to constrain the growth of the embedding space. Finally, despite the term "variational" in its name, the loss does not contain a variational loss objective because the prior $p(\mathbf{z})$ is assumed uniform and the KL-divergence term simplifies to a constant $\log K$.

The sg(.) in Eq. 3 and 4 is a stop-gradient operator defined as:

$$\text{sg}(\mathbf{x}) = \begin{cases} \mathbf{x}, & \text{forward pass} \\ \mathbf{0}, & \text{backward pass} \end{cases} \tag{5}$$

Our implementation of the VQ layer in PyTorch was validated with a number of unit tests and will be published with

the project code in a public repository as a pluggable layer for use in other network architectures[3].

3.3. Gumbel softmax

We also consider an alternative discrete latent variable model – a recently proposed Gumbel softmax latent distribution for VAEs outlined in (Maddison et al., 2016; Jang et al., 2016). The Gumbel sampling layer produces a set of one-hot vectors that describe samples from a discrete posterior distribution. This approach uses a reparameterisation trick on a Gumbel distribution, called the Gumbel Max trick. This is done in order to relax the discrete latent variables, such that $p(\mathbf{c}|\mathbf{x})$ becomes differentiable with respect to its parameters. Here, $\mathbf{c}$ is a vector of class probabilities $\alpha_1, \ldots, \alpha_k$ for one of the discrete latent variables.

A random variable G is distributed according to a standard Gumbel distribution if

$$G = -\log(-\log(U)) \tag{6}$$

for a uniform random variable $U \sim \text{Unif}[0, 1]$. We can then use an i.i.d. sequence of standard Gumbel random variables $g_1, \ldots, g_k$ with class probabilities α to obtain

$$c = \texttt{one\_hot}\left(\arg\max_k(\log \alpha_k + G_k)\right), \tag{7}$$

where we use a softmax to obtain a differentiable approximation to argmax. By transforming the samples according to

$$y_k = \frac{\exp((\log \alpha_k + g_k)\tau)}{\sum_i \exp((\log \alpha_k + g_k)\tau)}, \tag{8}$$

with τ corresponding to a temperature parameter to control the extent of the relaxation, we obtain a continuous and differentiable approximation to our discrete random variable.

3.4. Network architectures

All evaluated models use convolutional encoder and decoder networks with the only difference being the choice of the bottleneck layer: Gaussian sampling, VQ, or Gumbel softmax sampling, as described in the previous sections. In our experiments we employ two types of convolutions: a simple 1-dimensional convolution, and 1-dimensional gated convolution. In particular, the experiments using the spectral frame representation used simple convolutions, and in the experiments using the raw waveform we replace the convolutional layers with more powerful gated convolutions. Gated convolutions consist of two parallel convolutions, activated by sigmoid and hyperbolic tangent activation functions that are added together, as used in WaveNet architecture (van den Oord et al., 2016).

As an additional test, we compared model performance for both spectral frames and waveform data using the other respective architecture but found that the simpler architecture did not train on waveform data and that the more flexible architecture did not improve the results on spectral frames noticeably. As noted earlier, the state of the art for raw audio is currently the WaveNet architecture, but as this is computationally a very demanding architecture and not feasible for the present study, we resorted to a simpler alternative that still allows us to make relative comparisons between different latent representations. For details of the implementation and architectures see our repository https://github.com/vsimkus/voice-conversion.

We note that the speaker embedding is provided directly to the generator as shown in Figure 1, thus the encoder is allowed to filter out speaker-dependent features while keeping only the linguistic content. Voice conversion can then be achieved by providing the target speaker identity code to the generator and latent code $\mathbf{z}_g$ from the source speaker.

4. Experiments

We perform a full factorial experiment using VAE, VQ-VAE, and VAE with a latent Gumbel softmax distribution on both spectral frames and waveforms from the VCC2016 dataset (see Section 2.1). Here, we compare the ability to reconstruct speech and transfer the vocal style. We first assess the models' ability to reconstruct spectral frames in order to compare the different bottlenecks on a more structured data representation before assessing the models' capacity to reconstruct the more challenging waveform signals in the following section. Finally, we compare the quality of voice conversion using spectral frame features.

In our experiments, the encoder and generator architectures were kept constant between the different models in order to maintain comparability of the different approaches (for a detailed description, see Section 3.4). Therefore, only the bottleneck layer was changed according to the model. The convolutions used kernels of size 5 and 7 in alternating turns, *same* padding, and strides of 1 and 3 also in alternating turns, where the number of channels started at 32 and were increased by a factor of 2 every 2 layers, these choices were based on (Hsu et al., 2016). The Adam optimiser (Kingma & Ba, 2014) was used during training with a learning rate of 10^{-4}, and weight decay of 10^{-6}. Furthermore, when training on spectral frames, we used batch sizes of 512, and inputs of size 513 for a total number of 30 epochs or until convergence. When training on waveforms, we used batch sizes of 32, and inputs of size 8192 for a total number of 20 epochs or until convergence.

For exploring the dimensionality of the latent space, we use the following starting points, based on previous research. For the VAE model we used a 64-dimensional latent space, following Hsu et al. (2016). For the Gumbel VAE we used a 64-dimensional latent representation with 16 discrete values (for which no previous literature is available as a starting point). For VQ-VAE we used 512 discrete 64-dimensional embeddings following the authors' experiments in (van den Oord et al., 2017).

[3] https://github.com/vsimkus/voice-conversion/blob/master/models/vq_functions.py

Examples of reconstructed and voice-converted samples are provided here: https://soundcloud.com/user-479061401/sets/vc/s-0mS7E. Note that here we use the same original sample to allow for direct comparisons between methods, while we randomly sampled utterances for the evaluation study outlined in Section 2.4.

4.1. Spectral frame reconstruction

For spectral frames, the models were trained to minimise Gaussian log-probability loss $\hat{\mathbf{x}} \sim \mathcal{N}(\mathbf{x}, \mathbb{I})$ for the reconstruction term, where the mean of the Gaussian was the original input $\mathbf{x}$, and the covariance matrix was an identity (Hsu et al., 2016). The transformed spectral features were then combined with the original aperiodic features and F0, and synthesised into an audio sample using the WORLD vocoder, following (Hsu et al., 2016).

Preliminary experiments on the VQ-VAE have shown that the reconstruction quality was audibly indistinguishable from original samples. Hence, we have further tested the capacity of the VQ-VAE model by adding another downsampling convolutional layer (with a kernel size of 5 and stride of 3) to the encoder, and observed that the VQ-VAE was still able to perform comparably well under such a tight bottleneck. Additionally, we found that the model was able reconstruct intelligible outputs even when using only 2 discrete 4-dimensional latent embedding vectors. Therefore, in the further experiments we use this restricted VQ-VAE model, in order to force the model filter out more content-unrelated signal.

As presented in Figure 3, all models were capable of reconstructing spectral frames, while the quality of the reconstruction differed between the models. As indicated by our human evaluators (see Table 1), VQ-VAE was able to produce clearer reconstructions, compared to the standard VAE. The Gumbel VAE, however, performed worse than the VAE and VQ-VAE, as indicated by a descriptively lower average intelligibility score. As can be seen from the spectral envelope (Figure 3, lower right), the Gumbel VAE tended to produce more high-frequency signals (upper half of the spectral envelope) than the other methods, which people may have perceived as artefacts. Based on these results, it does not seem that the same level of performance of the VQ-VAE can be achieved by using the Gumbel VAE as a simpler representation of a discrete latent space, or a standard VAE with a continuous latent.

We conjecture that in the VQ-VAE, forcing the encoder to commit to the embedding dictionary acts as regularisation objective that encourages noise filtering and thus focuses more on the important features than the VAE. Similarly, we interpret these results such that the embedding vectors in VQ-VAE provides a richer information channel than the use of one-hot embeddings in the Gumbel VAE.

4.2. Raw waveform reconstruction

As a step towards fully end-to-end speech processing, we then evaluated our models on their capacity to reconstruct a raw waveform, encoded using only mu-law quantisation (as described in Section 2.2). Here, the reconstruction loss was given by the cross entropy loss between the quantised input samples and model outputs, which models were trained to minimise, following (van den Oord et al., 2017).

Out of the three models, only the VQ-VAE was able to reconstruct the original waveforms, whereas both VAE and Gumbel VAE produced flat outputs. This result concurs with the findings in Section 4.1. Specifically, the vector-quantisation objective enables the model to focus on the more descriptive features of the signal, thus allowing for an efficient representation of the waveform. Here, our results on the VQ-VAE align with findings from van den Oord et al. (2017), insofar as the VQ-VAE appears to be able to model global features of the waveform that span across many input dimensions, as argued in Chorowski et al. (2019).

However, we observe that for our VQ-VAE implementation, the produced reconstruction is far noisier for the waveform than for spectral frames. In Figure 4, we present one example of a reconstruction, as compared to the original sample. Even though the peaks of the signals match the original, showing that the VQ-VAE can reconstruct the basic shape of the signal, the quality is not comparable to that of the pre-processed spectral frames or to the samples produced using a WaveNet decoder in van den Oord et al. (2017). However, the finding that only the VQ-VAE was able to train on waveform data, gives support to its usefulness for learning representations of raw audio, as compared to a normal VAE or Gumbel VAE, which failed to train.

Therefore, we conjecture that for high-fidelity audio generation, our simplified architecture was not of sufficient capacity to allow for useful reconstructions. However, due to computational resource limitations, a WaveNet decoder architecture was not evaluated in this work.

4.3. Voice conversion using spectral frames

For assessing voice conversion, we only compare our models on spectral frames, as the audio reconstruction on waveform data did not have sufficient quality to allow for human evaluations. Here, the models were trained to minimise the same reconstruction loss as in Section 4.1. The speaker identity was passed directly to the decoder, where it was converted into a trainable speaker embedding and added to the latent representation before passing through the transpose-convolutional generator. Therefore, the encoder was encouraged to maintain the contents of the speech, but filter out speaker-related features, which were given to the generator "for free". The models were trained on all 10 speakers in the VCC2016 dataset. Hence, in this section we assess the voice conversion quality by reconstructing the source speaker's speech with a different target speaker's id.

During pre-processing, the mean and standard deviation of the fundamental frequency (F0) for each speaker were extracted from the dataset. For evaluation, the F0 was transformed between the source and target speakers using a linear mean-variance transformation on the log-F0 domain.

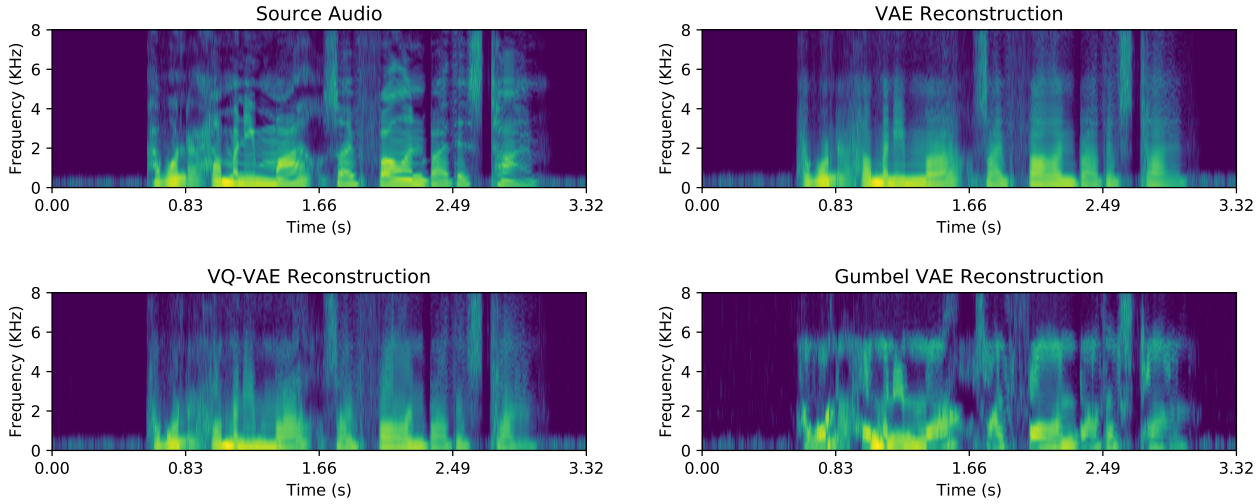

Figure 3. Spectral envelopes of the original audio sample, compared with reconstructions produced by each model.

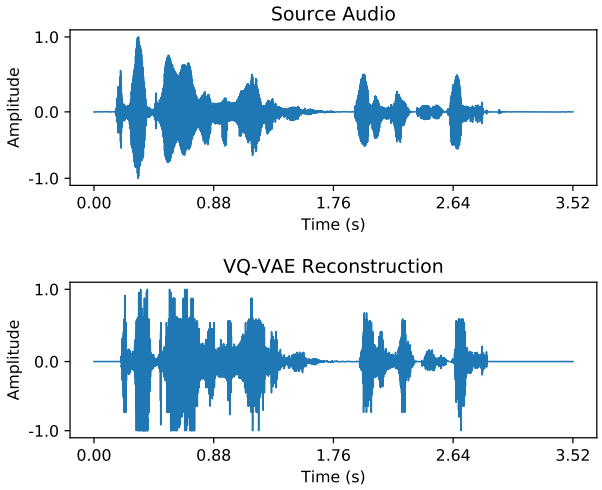

Figure 4. Waveforms produced by VQ-VAE compared with the source audio.

replicating Hsu et al. (2016). Finally, the reconstructed spectral features were combined with the aperiodic features and transformed F0, and synthesised using the WORLD vocoder.

People's evaluation showed that the similarity between a converted sample and the true sample as spoken by the target was highest for the standard VAE, while VQ-VAE and Gumbel VAE performed worse (as shown in Table 1). While the Gumbel VAE is rated slightly lower than the VQ-VAE, we hypothesise that this more due to the worse audio quality for the Gumbel VAE, rather a difference in VC ability.

To put our results into perspective, we evaluated our VAE model against the samples provided in Hsu et al. (2016; 2017) and noticed that our samples were comparable with the results presented in the related papers. In Figure 5, we use t-SNE on MFCC features (Mohamed et al., 2012), showing a clustering of reconstructions of utterances from the VAE model and voice converted samples. Here, we observe that the converted samples are clustered together with the target speaker's samples, indicating that VC could be achieved to some extent.

However, a further analysis of the underlying methodology has revealed that a significant factor of the effective voice conversion was due to the mean-variance transformation on log-F0, which was not reported in the original papers in Hsu et al. (2016; 2017). In particular, to evaluate the impact of the transformation on F0, we have synthesised samples using the transformed F0 and the original spectral frames and aperiodic features. As presented in Figure 5, we find that although the use of VAE-transformed spectral frames has an impact on the converted voice, a significant factor comes from the mean-variance transformation on F0. We also note that the F0-transformed samples were subjectively similar to the results published in the original paper on VAE-based voice conversion (Hsu et al., 2016; 2017).

Comparing our model's performances, we find that despite of the VQ-VAE model's descriptively better performance in spectral feature reconstruction, as described in Section 4.1, it performs worse at voice conversion than a regular VAE. We interpret this as being due to the lack of random posterior sampling, which may make the model slower to learn the speaker embeddings, but leave a formal evaluation to future research. Even though the Gumbel VAE does make use of sampling, we interpret its comparably poorer performance as being due to low-quality reconstruction performance, as described above.

5. Related work

While we focus on the latent space representation in the present report, another interesting frontier is given by exploring the way the reconstruction loss is computed. The

Method	Intelligibility		Similarity	
	Mean	*SD*	*Mean*	*SD*
VAE	2.50	0.90	3.0	1.04
VQ-VAE	3.42	0.79	2.17	0.58
Gumbel VAE	1.83	0.58	2.00	1.04

Table 1. Human evaluations of intelligibility and similarity, averaged across 12 participants. Ratings were provided on a 5-point scale, where higher values indicate more positive ratings.

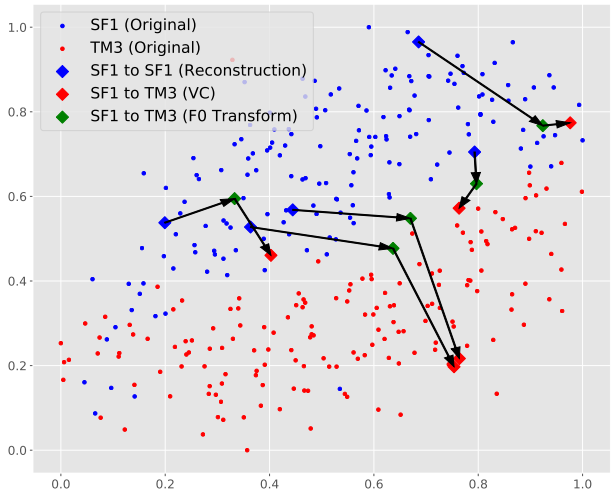

Figure 5. Clustering of voice samples using t-SNE on the MFCC statistics for two speakers. The smaller points represent the original samples in the dataset. The larger points represent the reconstructions of samples from speaker SF1 (blue) and the same samples voice converted to speaker TM3 (red) produced by the VAE model. The larger green points represent the audio generated from applying only the F0 transformation to the original sample.

output of VAEs in image and speech domains is typically blurry due to the element-wise l_2 reconstruction loss, which is equivalent to maximising the log-likelihood of data $p(\mathbf{x}|\hat{\mathbf{x}})$ and assuming it is Gaussian. However, real data such as images and speech are usually multimodal and therefore the Gaussianity assumption is too general. Here, future research may look further into replacing the reconstruction error with an adversarial loss, building on models such as VAE-GAN (Larsen et al., 2015) or AAE (Makhzani et al., 2015). For instance, one study by Hsu et al. (2017) has found that introducing an adversarial loss can enhance variability in the frequency axis, and hence improve voice clarity.

In addition, another possible improvement on the reconstruction might be achieved by employing a learnable feature-wise similarity metric method, which has previously been used in image processing (Larsen et al., 2015) to replace the element-wise metric in the VAE reconstruction loss. This might aid to further improve the intelligibility of the synthesised speech.

In van den Oord et al. (2017), the authors have shown that VQ-VAE can learn to classify phonemes in its latent layer with about 47% accuracy (random classification would have been about 7%) in a completely unsupervised manner. More recent research by Chorowski et al. (2019) looked into exploring the latent space provided by the VQ layer in comparison to a Gaussian VAE, where the VQ-VAE provided more accurate phoneme discovery. Whether this can also be observed when training on spectral frames is beyond the scope of the present report, as such analyses require training a separate phoneme classifier.

Furthermore, in Chorowski et al. (2019) the authors propose a dropout-inspired time-jitter regularisation. During training, the latent variables can replace one or both of its neighbours thus preventing token co-adaptation. We conjecture, that such regularisation scheme might encourage the use of speaker embeddings, similar to how random sampling in VAE, and thus increase the voice conversion capability.

6. Conclusions

The present results show that for reconstructing spectral frames, as obtained using a WORLD vocoder, the VQ-VAE model achieves a higher quality reconstruction than a standard VAE or a Gumbel VAE. When training on raw audio, only the VQ-VAE was able to train, while both other models could not learn a useful representation. Taken together, the findings indicate that while the discrete representation obtained from the VQ-VAE has advantages over a standard VAE with a continuous latent, the simpler alternative of using a Gumbel softmax does not yield the same results, and does in fact perform worse than a standard VAE on reconstruction.

Additionally, we find that reconstruction ability does not necessarily coincide with VC performance, as the VAE model was able to outperform both other models. Here, the Gumbel softmax performed similarly as the VQ-VAE, while obtaining poorer sounding reconstructions, as indicated by the human raters' evaluation.

Furthermore, we highlight that previous studies suffer from not assessing the effect that the transformation of the fundamental frequency has on VC, which should be taken into account by future investigations.

References

Abe, Masanobu, Nakamura, Satoshi, Shikano, Kiyohiro, and Kuwabara, Hisao. Voice conversion through vector quantization. *Journal of the Acoustical Society of Japan (E)*, 11(2):71–76, 1990.

Bengio, Yoshua, Nicholas, Leonard, and Courville, Aaron. Estimating or Propagating Gradients Through Stochastic Neurons for Conditional Computation. pp. 1–12, 2013.

Chorowski, Jan, Weiss, Ron J, Bengio, Samy, and Oord, Aäron van den. Unsupervised speech representation

learning using wavenet autoencoders. *arXiv preprint arXiv:1901.08810*, 2019.

Dupont, Emilien. Learning disentangled joint continuous and discrete representations. In *Advances in Neural Information Processing Systems*, pp. 708–718, 2018.

Fodor, JA, Bever, TG, and Garrett, M. *The psychology of language: An introduction to psycholinguistics*. New York: McGraw-Hill, 1974.

Goodfellow, Ian J., Pouget-Abadie, Jean, Mirza, Mehdi, Xu, Bing, Warde-Farley, David, Ozair, Sherjil, Courville, Aaron, and Bengio, Yoshua. Generative Adversarial Networks. In *Advances in Neural Information Processing Systems 27 (NIPS 2014)*, jun 2014. URL http://arxiv.org/abs/1406.2661.

Hsu, Chin-Cheng, Hwang, Hsin-Te, Wu, Yi-Chiao, Tsao, Yu, and Wang, Hsin-Min. Voice Conversion from Non-parallel Corpora Using Variational Auto-encoder. *arXiv e-prints*, pp. arXiv:1610.04019, oct 2016. URL http://arxiv.org/abs/1610.04019.

Hsu, Chin-cheng, Hwang, Hsin-te, Wu, Yi-chiao, Tsao, Yu, and Wang, Hsin-min. Voice Conversion from Unaligned Corpora using Variational Autoencoding Wasserstein Generative Adversarial Networks. (2), 2017.

Jang, Eric, Gu, Shixiang, and Poole, Ben. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

Karras, Tero, Laine, Samuli, and Aila, Timo. A style-based generator architecture for generative adversarial networks. *arXiv preprint arXiv:1812.04948*, 2018.

Kingma, Diederik P and Ba, Jimmy. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Kingma, Diederik P and Welling, Max. Auto-Encoding Variational Bayes. *arXiv e-prints*, pp. arXiv:1312.6114, dec 2013. URL http://arxiv.org/abs/1312.6114.

Kreiman, Jody and Sidtis, Diana. Foundations of Voice Studies: Introduction. In *Foundations of Voice Studies: An Interdisciplinary Approach to Voice Production and Perception*, chapter 1. John Wiley & Sons, 2011. ISBN 9781444395068. doi: 10.1002/9781444395068.

Larsen, Anders Boesen Lindbo, Sønderby, Søren Kaae, Larochelle, Hugo, and Winther, Ole. Autoencoding beyond pixels using a learned similarity metric. In *Proceedings of International Conference on Machine Learning (ICML)*, 2015.

Liu, Ming-Yu, Breuel, Thomas, and Kautz, Jan. Unsupervised image-to-image translation networks. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R. (eds.), *Advances in Neural Information Processing Systems 30*, pp. 700–708. Curran Associates, Inc., 2017. URL http://papers.nips.cc/paper/6672-unsupervised-image-to-image-translation-networks.pdf.

Lorenzo-Trueba, Jaime, Yamagishi, Junichi, Toda, Tomoki, Saito, Daisuke, Villavicencio, Fernando, Kinnunen, Tomi, and Ling, Zhenhua. The voice conversion challenge 2018: Promoting development of parallel and nonparallel methods. *arXiv preprint arXiv:1804.04262*, 2018.

Maddison, Chris J, Mnih, Andriy, and Teh, Yee Whye. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.

Makhzani, Alireza, Shlens, Jonathon, Jaitly, Navdeep, Goodfellow, Ian, and Frey, Brendan. Adversarial autoencoders. *arXiv preprint arXiv:1511.05644*, 2015.

Mohamed, Abdel-rahman, Hinton, Geoffrey, and Penn, Gerald. Understanding how deep belief networks perform acoustic modelling. *neural networks*, pp. 6–9, 2012.

Monaghan, Padraic and Christiansen, Morten H. Words in puddles of sound: Modelling psycholinguistic effects in speech segmentation. *Journal of child language*, 37(3): 545–564, 2010.

Morise, Masanori. Cheaptrick, a spectral envelope estimator for high-quality speech synthesis. *Speech Communication*, 67:1 – 7, 2015. ISSN 0167-6393. doi: https://doi.org/10.1016/j.specom.2014.09.003. URL http://www.sciencedirect.com/science/article/pii/S0167639314000697.

Morise, Masanori, Yokomori, Fumiya, and Ozawa, Kenji. World: a vocoder-based high-quality speech synthesis system for real-time applications. *IEICE TRANSACTIONS on Information and Systems*, 99(7):1877–1884, 2016.

Streijl, Robert C, Winkler, Stefan, and Hands, David S. Mean opinion score (mos) revisited: methods and applications, limitations and alternatives. *Multimedia Systems*, 22(2):213–227, 2016.

Toda, Tomoki, Chen, Ling Hui, Saito, Daisuke, Villavicencio, Fernando, Wester, Mirjam, Wu, Zhizheng, and Yamagishi, Junichi. The voice conversion challenge 2016. *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 08-12-Sept:1632–1636, 2016. ISSN 19909772. doi: 10.21437/Interspeech.2016-1066.

Traxler, Matthew and Gernsbacher, Morton Ann. *Handbook of psycholinguistics*. Elsevier, 2011.

van den Oord, Aaron, Dieleman, Sander, Zen, Heiga, Simonyan, Karen, Vinyals, Oriol, Graves, Alex, Kalchbrenner, Nal, Senior, Andrew, and Kavukcuoglu, Koray. WaveNet: A Generative Model for Raw Audio. sep 2016. URL http://arxiv.org/abs/1609.03499.

van den Oord, Aaron, Vinyals, Oriol, and Kavukcuoglu, Koray. Neural Discrete Representation Learning. In *Proceedings of Neural Information Processing Systems (NIPS 2017)*, 2017.

van der Maaten, Laurens and Hinton, Geoffrey. Visualizing Data using t-SNE. *Journal of Machine Learning Research*, 9(Nov):2579–2605, 2008. URL http://www.jmlr.org/papers/v9/vandermaaten08a.html.

Vondrick, Carl, Pirsiavash, Hamed, and Torralba, Antonio. Generating videos with scene dynamics. In *Advances In Neural Information Processing Systems*, pp. 613–621, 2016.