INTERSPEECH 2020

OCTOBER 25-29/ SHANGHAI , CHINA
SHANGHAI INTERNATIONAL CONVENTION CENTER

Transformer VQ-VAE for Unsupervised Unit Discovery and Speech Synthesis: ZeroSpeech 2020 Challenge

Andros Tjandra[1], Sakriani Sakti[1,2], Satoshi Nakamura[1,2]

[1] Nara Institute of Science and Technology, Japan

[2] RIKEN AIP, Japan

Outline

- Background
- Unsupervised subword discovery
  - VQVAE model & objective
  - Self-attention & Transformers
  - Model regularization
- Codebook inverter
- Experiment
- Conclusion

Background

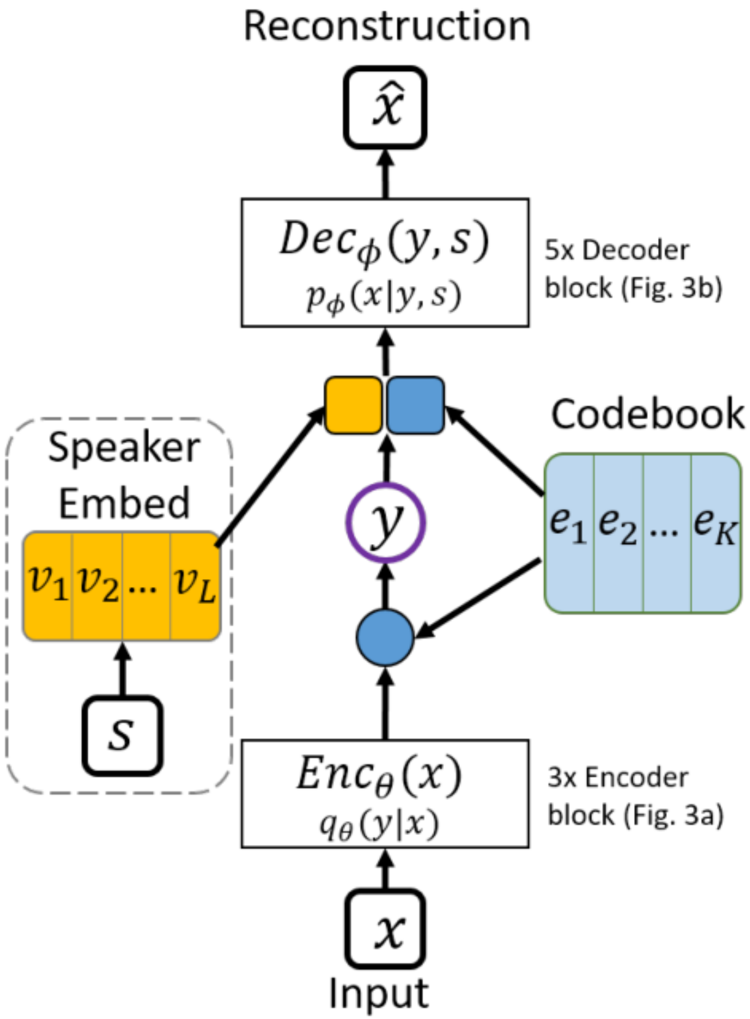
- The ZeroSpeech 2020 - Track 2019 challenge confronts the problem of constructing a speech synthesizer without any text or phonetic labels: TTS without T.

- Two objectives:
  - Find related contexts from the speech and encodes them as efficient as possible (low-bitrate).
  - Using the encoded representation, reconstruct the content back to the speech into another speaker voice.

Vector Quantized Variational Autoencoder (VQVAE)

- VQ-VAE has three main modules:
  - Encoder $q_\theta(z|x)$ read speech features $x \in \mathbb{R}^D$ and output $z \in \{1..K\}$
  - Codebook $E = [e_1, .., e_K] \in \mathbb{R}^{K \times D_e}$
  - Decoder $p_\phi(x|z, x)$ reconstruct the speech features conditioned by codebook z and speaker ID s

- Using explicit speaker information for the decoder, encoder and codebook only need to model the speech context without capturing the speaking style (disentangled with speaker).



Training Objective and Inference

- The discretization process in the encoder :

$$q_\theta(z = c|x) = \begin{cases} 1 & \text{if } c = \text{argmin}_i \|\hat{z} - e_i\|_2 \\ 0 & \text{else} \end{cases}$$

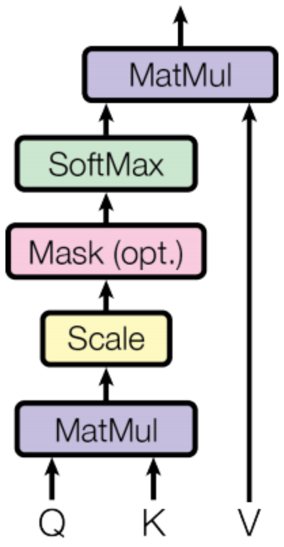
$$e_c = \sum_{i=1}^{K} q_\theta(z = i|x)\, e_i.$$

- Training objective:

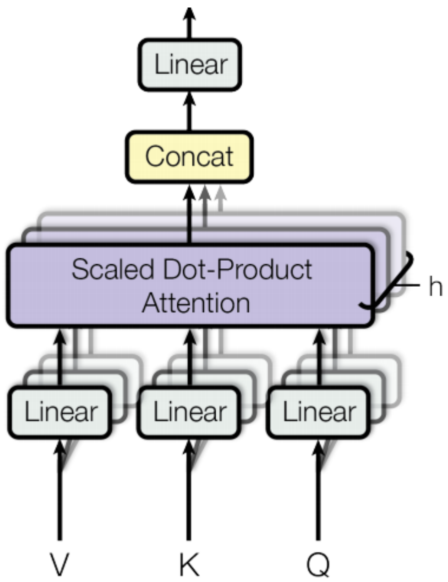
$$\mathcal{L}_{VQ} = \sum_{t=1}^{T} \underbrace{-\log p_\phi(x_t|y_t, s)}_{\text{Reconstruction loss}} + \underbrace{\gamma\|z_t - \text{sg}(e_{c_t})\|_2^2}_{\text{Consistency loss}}$$

Self Attention and Transformer

Dot Product Attention



Multihead Self Attention



Transformer module

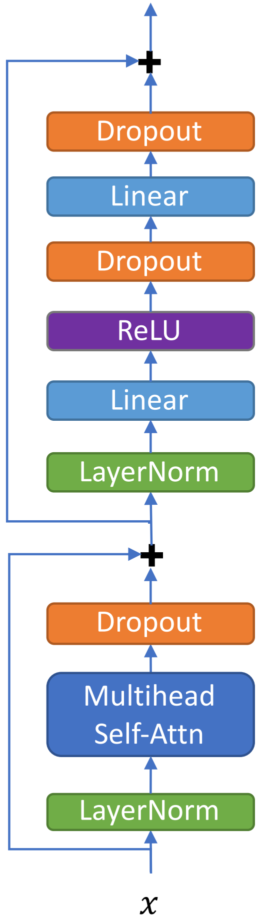
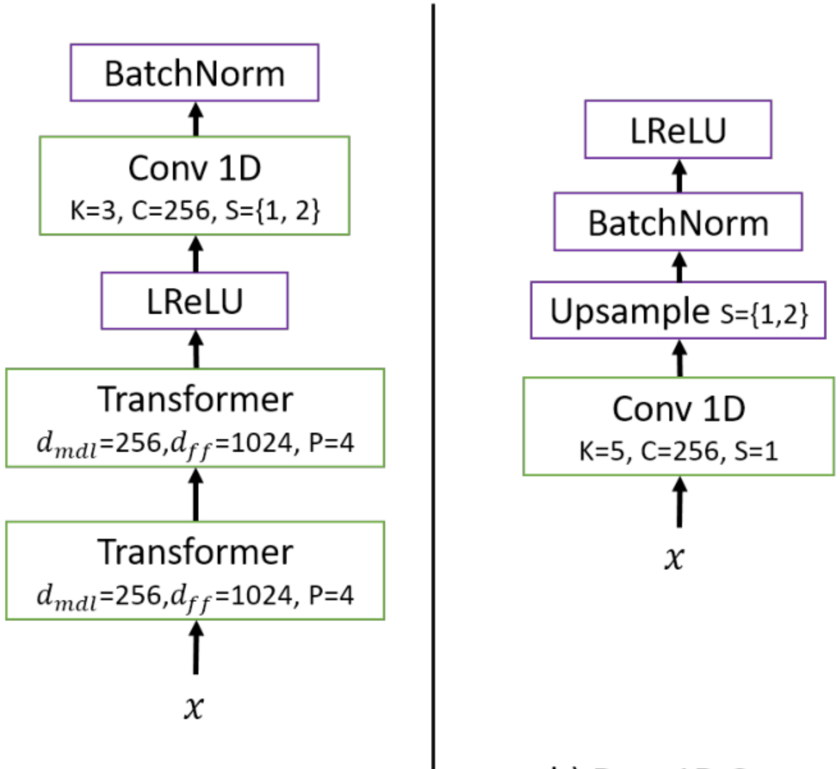


Image reference: Attention is all you need [Vaswani et al., NIPS 2017]

VQVAE Encoder & Decoder blocks

a) Encoder: 2x Transformer layer + 1D convolution (with stride to downsample input) + batchnorm

b) Decoder: 1D Convolution + Upsample operation + batchnorm



a) Enc: 2 Transformers + 1D Conv + Batchnorm

b) Dec: 1D Conv + Upsample + Batchnorm

Model regularization

- Sometimes, generative model could also suffer from overfitting especially when the amount of data is small.
- We deploy several regularization technique to improve the performance:
    1. Temporal smoothing
    2. Temporal jitter

Temporal Smoothing

- Since the encoder captures sequential data, we introduced temporal smoothing between two consecutive time-steps:

$$\mathcal{L}_{reg} = \sum_{i=1}^{T-1} \|z_t - z_{t+1}\|_2^2.$$

- Final loss:

$$\mathcal{L} = \mathcal{L}_{VQ} + \lambda \mathcal{L}_{reg},$$

Temporal Jitter

- Temporal jitter regularization [1] is used to prevent the latent vector co-adaptation and to reduce the model sensitivity near the unit boundary.

$$j_t \sim Categorical(p, p, 1 - 2 * p) \in \{1, 2, 3\}$$

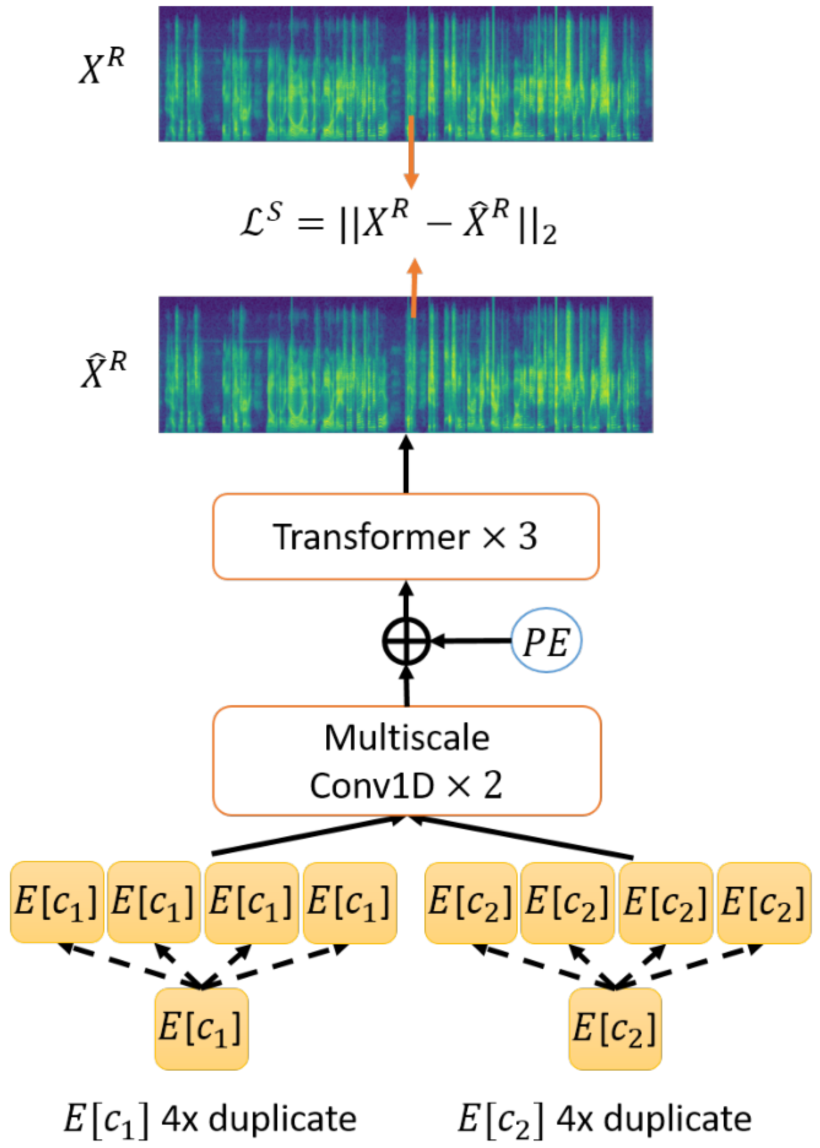
$$\hat{c_t} = \begin{cases} c_{t-1}, & \text{if } j_t = 1 \text{ and } t > 1 \\ c_{t+1}, & \text{if } j_t = 2 \text{ and } t < T \\ c_t, & \text{else} \end{cases}$$

$$e_t = \mathbf{E}[\hat{c_t}]$$

[1] Unsupervised speech representation learning using Wavenet autoencoders, [Chorowski et al., 2019]

Codebook Inverter

- Codebook inverter predicts the linear spectrogram given the predicted codebook from VQVAE.
- We use Griffin-Lim algorithm to generate the speech waveform
- Loss: $L^S = ||X^R - \hat{X}^R||_2$



Experimental Setup

- Log mel-spectrogram (80 dims) vs MFCC (39 dims with delta & delta-delta)

Model	ABX	Bitrate
TrfVQVAE with log-Mel	33.79	171.05
TrfVQVAE with MFCC	21.91	170.42

- For the rest of experiments, we will use MFCC features.

Conv VQVAE vs Transformer VQVAE

Model	ABX	Bitrate
Conv VQVAE (stride 4×, K=256) [1]	24.17	184.32
TrfVQVAE (stride 4×, K=64)	22.72	141.82
TrfVQVAE (stride 4×, K=128)	21.91	170.42
TrfVQVAE (stride 4×, K=256)	21.94	194.69
TrfVQVAE (stride 4×, K=512)	21.6	217.47

-2.2 ABX

[1] VQVAE Unsupervised Unit Discovery and Multi-Scale Code2Spec Inverter for Zerospeech Challenge 2019 [Tjandra et al, 2019]

Best model + regularization

Model	ABX	Bitrate
TrfVQVAE (stride $4\times$, K=128)	21.91	170.42
+ temp smooth ($\lambda = 1e-2$)	21.88	169.02
+ temp smooth ($\lambda = 5e-3$)	21.67	169.2
+ temp smooth ($\lambda = 1e-3$)	21.75	169.56
+ temp jitter ($p = 0.05$)	21.57	166.19
+ temp jitter ($p = 0.075$)	21.70	164.08
+ temp smooth ($\lambda = 5e-3$) + temp jitter ($p = 0.05$)	20.71	171.99
+ temp smooth ($\lambda = 1e-3$) + temp jitter ($p = 0.05$)	20.14	167.02

-0.24 ABX

-0.34 ABX

-1.77 ABX

Conclusions

- We achieved significant improvement with Transformer VQ-VAE (-2.2 ABX vs Conv VQ-VAE).
- By adding regularization in the VQVAE encoder and codebook, we get further improvement (up to -1.77 ABX vs un-regularized model.)

The end.