INTERSPEECH 2020
October 25–29, 2020, Shanghai, China

Exploring TTS without T Using Biologically/Psychologically Motivated Neural Network Modules (ZeroSpeech 2020)

Takashi Morita, Hiroki Koda

Primate Research Institute, Kyoto University, Japan
tmorita@alum.mit.edu, koda.hiroki.7a@kyoto-u.ac.jp

Abstract

In this study, we reported our exploration of Text-To-Speech without Text (TTS without T) in the Zero Resource Speech Challenge 2020, in which participants proposed an end-to-end, unsupervised system that learned speech recognition and TTS together. We addressed the challenge using biologically/psychologically motivated modules of Artificial Neural Networks (ANN), with a particular interest in unsupervised learning of human language as a biological/psychological problem. The system first processes Mel Frequency Cepstral Coefficient (MFCC) frames with an Echo-State Network (ESN), and simulates computations in cortical microcircuits. The outcome is discretized by our original Variational Autoencoder (VAE) that implements the Dirichlet-based Bayesian clustering widely accepted in computational linguistics and cognitive science. The discretized signal is then reverted into sound waveform via a neural-network implementation of the source-filter model for speech production.

Index Terms: unsupervised learning, attention mechanism, Dirichlet-Categorical distribution, echo-state network, source-filter model

1. Introduction

Recent developments in ANNs have drastically improved various tasks in natural language processing. However, these developments are based on a large amount of annotated data, which are not available with respect to many minority languages and for real language-learning children. Accordingly, unsupervised learning of languages based on raw speech data is required for industrial and academic purposes. This study reports our exploration of TTS without T in the Zero Resource Speech Challenge 2020, which intended to develop an end-to-end, unsupervised system that can learn speech recognition and TTS together. Even though the participants in the previous challenge mainly investigated mechanically sophisticated systems [1, 2], we addressed the challenge using biologically/psychologically motivated ANN modules, by considering the unsupervised learning of human language as a biological/psychological problem. One of the adopted modules is our original discrete VAE that implements the Dirichlet-based Bayesian clustering within the end-to-end system.

2. System Description

Our network consisted of the auditory module (encoder), the symbolic module (discrete VAE), and the articulatory module (decoder; see Fig. 1), each of which is described in details, below. The dimensionality of the hidden layers of the network is 128, unless otherwise specified.

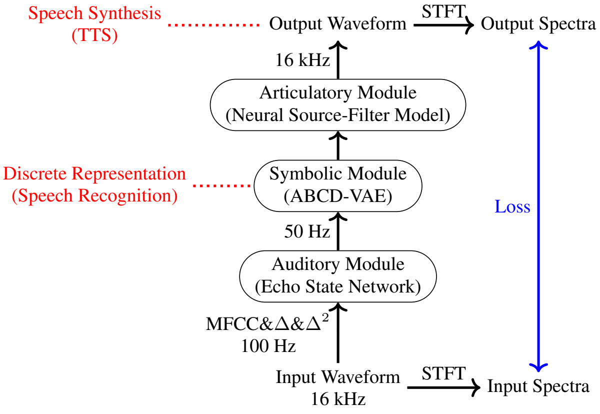

Speech Synthesis (TTS)
Output Waveform
STFT
Output Spectra
16 kHz
Articulatory Module (Neural Source-Filter Model)
Discrete Representation (Speech Recognition)
Symbolic Module (ABCD-VAE)
Loss
50 Hz
Auditory Module (Echo State Network)
MFCC&Δ&Δ^2
100 Hz
Input Waveform 16 kHz
STFT
Input Spectra

Figure 1: *Overall architecture.*

2.1. Auditory Module: ESN

The waveform data are first converted into 13 MFCCs (with 25 msec Hanning window and 10 msec stride), with their first and second derivatives concatenated. Those frames are then processed by an ESN (with 2048 neurons) [3]. ESNs are recurrent neural networks having sparse hidden-to-hidden connections (10%), which are randomly initialized and fixed without updates via learning. ESNs and their related model, liquid state machine, simulate computations in cortical microcircuits [4]. We took every second output of the ESN and downsampled the signals from 100 Hz to 50 Hz.

2.2. Symbolic Module: ABCD-VAE

The output from the auditory module is a time series of real-valued vectors, and the role of the symbolic module is to discretize them. Ideally, this module should classify sound frames into linguistically meaningful categories like phonemes whose utterance spans over the frames. Since the size of phonemic inventory varies among languages, the symbolic module should also detect the appropriate number of categories instead of just putting frames into a predetermined number of classes. A popular approach to such clustering problems in computational linguistics and cognitive science is Bayesian clustering based on the Dirichlet distribution/process [6, 7, 8, 9, 10, 11, 12, 13, 14]. To build this Dirichlet-based clustering in our end-to-end system, we propose a novel discrete VAE named the *ABCD-VAE*, whose first four letters stand for the Attention-Based Categorical sampling with the Dirichlet prior (simultaneously explored for analysis of birdsong in [15]). The ABCD-VAE converts each output frame $\mathbf{x}_i$ from the auditory module to the probability $q(z_i \mid \mathbf{x}_i)$ of its classification to a discrete category z_i. This mapping from $\mathbf{x}_i$ to $q(z_i \mid \mathbf{x}_i)$ is implemented by a

http://dx.doi.org/10.21437/Interspeech.2020-3127

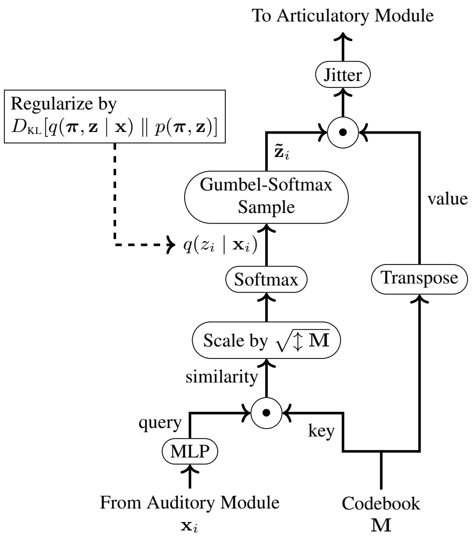


Figure 2: *ABCD-VAE, implemented as the scaled dot-product attention mechanism. The symbol "$\updownarrow \mathbf{M}$" represents the number of rows in the codebook matrix $\mathbf{M}$ (i.e., the dimensionality of the column vectors, set as 128) [5].*

Multi-Layer Perceptron (MLP) followed by a linear transform.

The prior on z_i is the Dirichlet-Categorical distribution:

$$\boldsymbol{\pi} := (\pi_1, \dots, \pi_K) \sim \text{Dirichlet}(\boldsymbol{\alpha}) \tag{1}$$

$$z_i \mid \boldsymbol{\pi} \sim \text{Categorical}(\boldsymbol{\pi}) \tag{2}$$

where the concentration $\boldsymbol{\alpha} := (\alpha_1, \dots, \alpha_K)$ is a free parameter and we set it as $\alpha_k = 1, \forall k \in \{1, \dots, K\}$; $K = 256$ is an upper-bound for the number of frame categories to be used.

The classification probability $q(z_i \mid \mathbf{x}_i)$ is used to sample a one-hot-like vector $\tilde{\mathbf{z}}_i$ that approximates the categorical sample by the Gumbel-Softmax distribution [16]. The output of the ABCD-VAE is a continuous-valued vector given by $\tilde{\mathbf{z}}_i \mathbf{M}^{\mathrm{T}}$, where $\mathbf{M}$ is a learnable matrix and $\tilde{\mathbf{z}}_i$ picks up one of its column vector, if it is indeed a one-hot vector. One implementational trick here is that $\mathbf{M}$ is shared with the final linear transform yielding the classification probability $q(z_i \mid \mathbf{x}_i)$. Accordingly, the classification logits are given by the dot-product similarities between the transformation of $\mathbf{x}_i$ and the column vectors of $\mathbf{M}$. This links the probability computation and the VAE's output just as the Vector-Quantized-VAE (VQ-VAE) does with the L2 distance [17, 18], and makes the learning easier. This shared-matrix architecture can be seen as the attention mechanism (having identical key and value) [5, 19, 20] and, thus, we call it the attention-based categorical sampling (Fig. 2).

We want to classify contiguous frames into the same category except when they include a phonemic boundary. To encourage such classification, we adopted the jitter regularizer that randomly replaces each frame's category with one of its adjacent frames with probability, 0.12 [18].

2.3. Articulatory Module: Neural Source-Filter Model

The articulatory module receives the output from the symbolic module, and produces waveform from it. A physiologically motivated model for the human voice production is the source-filter model. We adopted a neural-network implementation of the source-filter model for the articulatory module (specifically, the hn-NSF in [21]; see Fig. 3).

Table 1: *Short-Time Fourier Transform (STFT) configurations of the spectral loss.*

FFT Config. Type	FFT Bins	Frame Length	Stride
Time-Dedicated	128	80 (5 ms)	40 (2.5 ms)
Same as Input	512	400 (5 ms)	100 (5 ms)
Freq.-Dedicated	2048	1920 (120 ms)	640 (40 ms)

The articulatory module first processed the output from the ABCD-VAE with 3 layers of the bidirectional Long Short-Term Memory (LSTM), conditioned on two speaker embeddings; one of them is used as the initial hidden state and the other is concatenated with each frame of the output from the symbolic module. The module then upsampled the outcome with four layers of transposed convolution (whose stride and kernel size were 5 and 25, respectively, for the bottom layer, and 4 and 16, respectively, elsewhere; the hidden dimensionality of the LSTM and transposed convolution was 128). This yielded a 16 kHz sequence with 64 channels, which is termed $c_{j,t}$ in Fig. 3.

The first channel of the upsampled sequence, $c_{1,1}, \dots, c_{1,T}$, is intended to represent the base frequency of the output waveform in the log scale. This F0-like channel is fed to the *source* submodule that generated excitation signals from harmonic sine waves. The *filter* submodule transformed the resulting signals through 5 blocks of 10-layer dilated convolution (with 64 channels), conditioned on $c_{j,t}$. The articulatory module had another source-filter flow designed for the production of noisy sounds, such as fricative consonants. The noisy source is produced by a Gaussian and filtered through a single block of 10-layer dilated convolution. (See [21] for the detailed architecture of the source and filter submodules.)

The harmonic and noisy outcomes are transformed by low-pass and high-pass Finite Impulse Response (FIR) filters, respectively, and combined additively. We made two pairs of filters, one specialized for voiced sounds (0-5 kHz low-pass/stop and 7-8 kHz high-pass/stop bands) and the other for voiceless sounds (0-1 kHz low-pass/stop and 3-8 kHz high-pass/stop bands). The filter coefficients are computed using the Remez exchange algorithm (with the order 10) [22, 23]. The use of the voiced vs. voiceless sounds is switched by the F0-like signal; F0 $\gg 0$ flags the voiced sounds and F0 ≈ 0 flags the voiceless sounds.

2.4. Training Objectives

The training objective, $\mathcal{L}$, of the network is given by, $\mathcal{L} := \bar{\mathcal{L}}_{\text{SPEC}} + \mathcal{L}_{\text{KL}}$, which clearly consists of two terms. The first term, $\bar{\mathcal{L}}_{\text{SPEC}}$, compared the spectra of the input and output signals with three different time-frequency resolutions (Table 1). The spectral loss between the input and output spectral sequences with L frames and M frequencies, $\mathcal{L}_{\text{SPEC}}^{(M,L)}$ is defined by [21]:

$$\mathcal{L}_{\text{SPEC}}^{(M,L)} := \frac{1}{2LM} \sum_{l=1}^{L} \sum_{m=1}^{M} \left(\log \frac{|y_{m,l}|^2 + \epsilon}{|\hat{y}_{m,l}|^2 + \epsilon} \right)^2 \tag{3}$$

where $y_{m,l}$ and $\hat{y}_{m,l}$ are the input and output spectral sequences, respectively, and $\epsilon = 10^{-5}$. The average of $\mathcal{L}_{\text{SPEC}}^{(M,L)}$ over the different time-frequency resolutions yielded the $\bar{\mathcal{L}}_{\text{SPEC}}$.

The second term of the training objective, $\mathcal{L}_{\text{KL}}$, regularized the ABCD-VAE by the Kullback-Leibler (KL) divergence of the posterior probability, $q(\boldsymbol{\pi}, \mathbf{z})$ to the prior $p(\boldsymbol{\pi}, \mathbf{z})$ (cf. [24]). To make this KL divergence computable, we adopted the mean-field variational inference and assumed that (i) $\boldsymbol{\pi}$ is independent

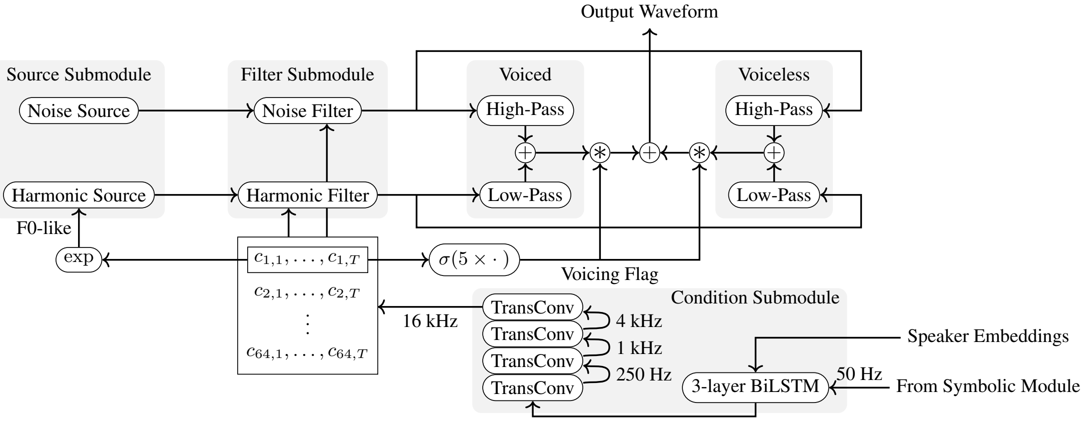


Figure 3: *The neural source-filter model, which customized the hn-NSF model in [21].*

of $\mathbf{z}$ and $\mathbf{x}$ in q and (ii) $q(\boldsymbol{\pi})$ is a Dirichlet distribution [25]. Then, the KL divergence is rewritten as follows:

$$\mathcal{D}_{\mathrm{KL}}\left[q(\boldsymbol{\pi},\mathbf{z}) \parallel p(\boldsymbol{\pi},\mathbf{z})\right] = \mathcal{D}_{\mathrm{KL}}\left[q(\boldsymbol{\pi}) \parallel p(\boldsymbol{\pi})\right] + \sum_{i=1}^{N} \mathcal{D}_{z_i} \quad (4)$$

$$\mathcal{D}_{\mathrm{KL}}\left[q(\boldsymbol{\pi}) \parallel p(\boldsymbol{\pi})\right] = \mathrm{E}_q\left[\log q(\boldsymbol{\pi})\right] - \mathrm{E}_q\left[\log p(\boldsymbol{\pi})\right] \quad (5)$$

$$\mathcal{D}_{z_i} := \mathrm{E}_q\left[\log q(z_i \mid \mathbf{y}_i)\right] - \mathrm{E}_q\left[\log p(z_i \mid \boldsymbol{\pi})\right] \quad (6)$$

where all the expectations have a closed form.

The frame-based classification by the ABCD-VAE can result in an undesirable situation in which only categories spanning over many frames are detected, and short segments such as consonants are ignored due to the Occam's razor effect of the Dirichlet, prior. To address this issue, we divided each $\mathcal{D}_{z_i}$ by U_i, the number of contiguous frames, including the one indexed by i, that yielded the same most probable category. This weighting applied the Dirichlet-based clustering to spans of frames instead of individual frames, which ideally correspond to phonetic segments. Accordingly, $\mathcal{L}_{\mathrm{KL}}$ is defined for each data sequence as follows:

$$\mathcal{L}_{\mathrm{KL}} := \frac{1}{T}\left(\frac{S}{N}\mathcal{D}_{\mathrm{KL}}\left[q(\boldsymbol{\pi}) \parallel p(\boldsymbol{\pi})\right] + \sum_{i=1}^{S}\frac{1}{U_i}\mathcal{D}_{z_i}\right) \quad (7)$$

where T is the number of samples in the waveform, S is the number of frames in the discrete representation of the sequence, and N is the total number of frames in the whole dataset.

The concentration parameter, $\boldsymbol{\omega} := (\omega_1,\ldots,\omega_K)$ of the Dirichlet, $q(\boldsymbol{\pi})$ is optimized by:

$$\omega_k = \alpha_k + \sum_{i=1}^{N} q(z_i = k \mid \mathbf{x}_i) \quad (8)$$

$$= \alpha_k + N\theta_k \quad (9)$$

The second term on the right-hand side of Eq. 8 requires summation over all the N frames, across different minibatches, and this is not efficient in minibatch learning. Accordingly, we adopted Eq. 9, instead of Eq. 8, where θ_k are learnable parameters of the ABCD-VAE such that $\sum_k \theta_k = 1$.

We trained the network for 36k iterations, using the Adam optimizer ($\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$) [26]. The learning rate is initially set as 4×10^{-4} and halved at 16k, 24k, and 32k iterations. During the first 4k iterations, we did not sample from the Gumbel-Softmax distribution in the ABCD-VAE, instead, we directly multiplied the classification probabilities, $q(\cdot \mid \mathbf{x}_i)$, and the transposed codebook, $\mathbf{M}^{\mathrm{T}}$. After this "pretraining" phrase, we annealed the temperature, τ, of the Gumbel-Softmax following the equation, $\tau = \max\{0.5, \exp(-10^{-5}\iota)\}$, at an interval of 1k iterations, where ι counts the iterations [16]. Each batch consisted of 16 segments of speech sound whose duration was 1 s or shorter. Those speech segments are randomly selected from the entire wav files when the files were longer than 1 s. The network is implemented in PyTorch,[1] and ran on a private server with a single NVIDIA GeForce RTX 2080Ti graphic card.

3. Results

The model performance for the speech recognition task (encoding) is evaluated on the ABX discriminability and bitrate of the embeddings. The ABX discriminability scored the model's ability to distinguish phonetic minimal pairs; The model is ran on minimal-paired signals, A and B, and another signal from a different speaker, X, whose gold-standard transcription was identical to that of A. The ABX error rate is given by the probability that the model incorrectly assigned closer latent representation to B and X than to A and X. The bitrate evaluated the compression in the latent representation.

The ABX error rate of our proposed system was 39.30 for English and 34.41 for the surprise language [27, 28], when the *Maximum a Posteriori* (MAP) classification is scored using the Levenshtein distance and contiguous classmate frames are merged (Table 2). The error rates decreased to 35.46 and 18.87, respectively when the posterior probability (of the first frame of each classmate span) is scored using the KL divergence and the dynamic time warping (submitted as "Auxiliary Embedding 1"). These posterior-based scores are comparable to the baseline scores [1, 29, 30], and even to the topline score of the surprise language [1]. The gap between the MAP-based and posterior-based scores indicates that the MAP classification often ignored small acoustic differences in minimal pairs and their discrimination needed reference to non-MAP categories. For example, the model may assign the same MAP category to [ɪ] and [ɛ] but their second probable category may be different, say [i] vs. [e].

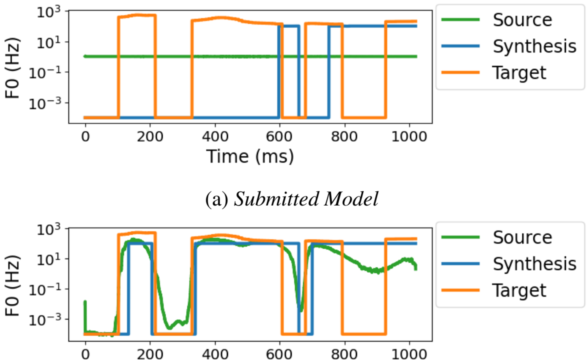
[1] The code used in this study is available in `https://github.com/tkc-morita/ZeroSpeech2020_TTSwoT.git`.

Table 2: *Scores for the speech recognition (encoding).*

Language	Score Type	Baseline	Topline	Submitted Model		CNN Encoder		+F0 Learning	
				MAP	Posterior	MAP	Posterior	MAP	Posterior
English	ABX	35.63	29.85	39.30	35.46	38.76	34.57	42.98	41.68
	Bitrate	71.98	37.73	137.58	405.37	187.12	548.73	105.56	327.40
Surprise	ABX	27.46	16.09	34.41	18.87	—	—	—	—
	Bitrate	74.55	35.20	151.03	—	—	—	—	—

Table 3: *Scores for the TTS (decoding).*

Language	Score Type	Baseline	Topline	Submitted Model
English	MOS	2.14	2.52	1.19
	CER	0.77	0.43	0.67
	Similarity	2.98	3.10	1.14
Surprise	MOS	2.23	3.49	1.77
	CER	0.67	0.33	0.46
	Similarity	3.26	3.77	1.22



(a) *Submitted Model*

(b) *+F0 Learning*

Figure 4: *Contours of F0 in the input of the source submodule (green), the speech synthesis (blue), and the target voice (orange). We used* `V002_4165536801.wav` *in the voice dataset. F0 was zero when the speech sound was voiceless.*

The simplest and, thus, possibly the weakest module of our model is the auditory module, which consisted of the ESN, or a random, non-trainable Recurrent Neural Network (RNN). To evaluate this module, we replaced the ESN with a Convolutional Neural Network (CNN)-based encoder adopted in [18] and tested it on the English data. This replacement made only small improvements in the ABX scores, in compensation for the increased bitrates.

The TTS performance is evaluated by three human-judged scores: Mean Opinion Score (MOS) on speech synthesis (greater values are better), Character Error Rate (CER) after human transcription of speech synthesis (the smaller, the better), and similarity to the target voice of speech synthesis (the greater, the better). Our model got better CER scores than the baseline (Table 3), indicating that the encoder extracted important linguistic features and the decoder recovered them in the speech synthesis. In contrast, the MOS and the similarity scores of our model were worse than the baseline. We found that one major problem with the model was that it failed to learn the F0-like features used as the input to the source submodule (Fig. 4a), making the synthesized voice sound robotic. This particular problem can be remedied by including the F0 loss in the objective function (Fig. 4b; the mean squared error between the F0 feature and ground truth is measured, and $\bar{\mathcal{L}}_{\text{SPEC}}$ is replaced with the average of the F0 loss and it; F0 is computed using Persel-mouth, a Python wrapper of Praat). However, this F0 learning degraded the ABX scores (Table 2), and the speech synthesis still sounded robotic to the authors.

4. Discussion

This study explored the TTS-without-T task using biologically/psychologically motivated modules of neural networks: the ESN for the auditory module, the ABCD-VAE for the symbolic module, and the neural source-filter model for the articulatory module. Our technical contribution is the ABCD-VAE. It implemented the Dirichlet-based clustering in neural networks and made the end-to-end system possible. Specifically, it automatically detects the statistically optimal number of frame categories (under an arbitrary upper bound) and enables more non-parametric learning than other discrete VAEs [16, 17, 18].

The ABCD-VAE yielded linguistically informative representation in the posteriorgrams, but the MAP representation missed some of this information. This failure is likely not because of the limited capacity of the ESN encoder, which did not have any learnable parameters, since the canonical CNN-based encoder, adopted in previous work, yielded similar scores. Similar problems were reported in last year's Zero Resource Speech Challenge; most of the proposed discrete representations—particularly those with low bitrates—exhibited higher ABX error rates than the baseline, indicating a general difficulty in unsupervised learning of such representations [1].

A bigger problem is found in the articulatory module—failure to learn the F0 feature used in the source submodule. The naive learning of F0 improved this particular feature, but degraded the discrete representation of the ABCD-VAE and did not make the synthesized voice robustly more natural. (Note that the original study of the neural source-filter model fed the gold-standard F0 to the model and thus the F0 learning was not an issue.) Thus, successful training of this articulatory module—in the end-to-end setting—will be the central issue for the next Zero Resource Speech Challenge in our framework.

5. Acknowledgements

We gratefully acknowledge the support of MEXT Grant-in-aid for Scientific Research on Innovative Areas #4903 (Evolinguistic; JP17H06380) and the JST Core Research for Evolutional Science and Technology 17941861 (#JPMJCR17A4).

6. References

[1] E. Dunbar, R. Algayres, J. Karadayi, M. Bernard, J. Benjumea, X.-N. Cao, L. Miskic, C. Dugrain, L. Ondel, A. W. Black, L. Besacier, S. Sakti, and E. Dupoux, "The Zero Resource Speech Challenge 2019: TTS without T," in *Proceedings of Interspeech 2019*, 2019, pp. 1088–1092.

[2] A. Tjandra, B. Sisman, M. Zhang, S. Sakti, H. Li, and S. Nakamura, "VQVAE unsupervised unit discovery and multi-scale Code2Spec inverter for Zerospeech Challenge 2019," 2019.

[3] H. Jaeger and H. Haas, "Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication," *Science*, vol. 304, no. 5667, pp. 78–80, 2004.

[4] W. Maass, T. Natschläger, and H. Markram, "Real-time computing without stable states: A new framework for neural computation based on perturbations," *Neural Computation*, vol. 14, no. 11, pp. 2531–2560, 2002.

[5] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds. Curran Associates, Inc., 2017, pp. 5998–6008.

[6] J. R. Anderson, *The adaptive character of thought*, ser. Studies in cognition. Hillsdale, NJ: L. Erlbaum Associates, 1990.

[7] K. Kurihara and T. Sato, "An application of the variational Bayesian approach to probabilistic context-free grammars," in *International Joint Conference on Natural Language Processing Workshop Beyond Shallow Analyses*, 2004.

[8] ——, "Variational Bayesian grammar induction for natural language," in *Grammatical Inference: Algorithms and Applications: 8th International Colloquium, ICGI 2006, Tokyo, Japan, September 20-22, 2006. Proceedings*, Y. Sakakibara, S. Kobayashi, K. Sato, T. Nishino, and E. Tomita, Eds. Berlin, Heidelberg: Springer Berlin Heidelberg, 2006, pp. 84–96.

[9] Y. W. Teh, M. I. Jordan, M. J. Beal, and D. M. Blei, "Hierarchical Dirichlet processes," *Journal of the American Statistical Association*, vol. 101, no. 476, pp. 1566–1581, 2006.

[10] C. Kemp, A. Perfors, and J. Tenenbaum, "Learning overhypotheses with hierarchical Bayesian models," *Developmental Science*, vol. 10, no. 3, pp. 307–321, 2007.

[11] S. Goldwater, T. L Griffiths, and M. Johnson, "A Bayesian framework for word segmentation: Exploring the effects of context," *Cognition*, vol. 112, pp. 21–54, 04 2009.

[12] N. H. Feldman, S. Goldwater, T. L. Griffiths, and J. L. Morgan, "A role for the developing lexicon in phonetic category acquisition," *Psychological Review*, vol. 120, no. 4, pp. 751–778, 2013.

[13] H. Kamper, A. Jansen, and S. Goldwater, "A segmental framework for fully-unsupervised large-vocabulary speech recognition," *Computer Speech & Language*, vol. 46, pp. 154–174, 2017.

[14] T. Morita and T. J. O'Donnell, "Statistical evidence for learnable lexical subclasses in Japanese," *Linguistic Inquiry*, To appear, doi:10.1162/ling_a_00401.

[15] T. Morita, H. Koda, K. Okanoya, and R. O. Tachibana, "Measuring long context dependency in birdsong using an artificial neural network with a long-lasting working memory," *bioRxiv*, 2020, doi:10.1101/2020.05.09.083907.

[16] E. Jang, S. Gu, and B. Poole, "Categorical reparameterization with Gumbel-softmax," in *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*, 2017.

[17] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, "Neural discrete representation learning," in *Advances in Neural Information Processing Systems 30*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds. Curran Associates, Inc., 2017, pp. 6306–6315.

[18] J. Chorowski, R. J. Weiss, S. Bengio, and A. van den Oord, "Unsupervised speech representation learning using wavenet autoencoders," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 12, pp. 2041–2053, Dec 2019.

[19] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[20] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," arXiv:1810.04805, 2018.

[21] X. Wang, S. Takaki, and J. Yamagishi, "Neural source-filter waveform models for statistical parametric speech synthesis," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 28, pp. 402–415, 2020.

[22] J. H. McClellan and T. W. Parks, "A unified approach to the design of optimum FIR linear-phase digital filters," *IEEE Trans. Circuit Theory*, vol. 20, no. 6, pp. 697–701, 1973.

[23] J. H. McClellan, P. T. W., and L. R. Rabiner, "A computer program for designing optimum fir linear phase digital filters," *IEEE Trans. Audio and Electroacoustics*, vol. 21, pp. 506–526, 1973.

[24] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," The International Conference on Learning Representations (ICLR) 2014, 2014.

[25] C. M. Bishop, *Pattern recognition and machine learning*, ser. Information science and statistics. New York: Springer, 2006.

[26] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[27] S. Sakti, R. Maia, S. Sakai, T. Shimizu, and S. Nakamura, "Development of HMM-based Indonesian speech synthesis," in *Proceedings of O-COCOSDA*, 2008, pp. 215–220.

[28] S. Sakti, E. Kelana, H. Riza, S. Sakai, K. Markov, and S. Nakamura, "Development of Indonesian large vocabulary continuous speech recognition system within A-STAR project," in *Proceedings of the Workshop on Technologies and Corpora for Asia-Pacific Speech Translation (TCAST)*, 2008, pp. 19–24.

[29] L. Ondel, L. Burget, and J. Černocký, "Variational inference for acoustic unit discovery," *Procedia Computer Science*, vol. 81, pp. 80–86, 2016, sLTU-2016 5th Workshop on Spoken Language Technologies for Under-resourced languages 09-12 May 2016 Yogyakarta, Indonesia.

[30] Z. Wu, O. Watts, and S. King, "Merlin: An open source neural network speech synthesis system," in *9th ISCA Speech Synthesis Workshop*, 2016, pp. 202–207.