The Zero Resource Speech Challenge 2019: TTS without T

Ewan Dunbar\textsuperscript{1,2{*}}, Robin Algayres\textsuperscript{2}, Julien Karadayi\textsuperscript{2}, Mathieu Bernard\textsuperscript{2}, Juan Benjumea\textsuperscript{2}, Xuan-Nga Cao\textsuperscript{2}, Lucie Miskic\textsuperscript{1}, Charlotte Dugrain\textsuperscript{1}, Lucas Ondel\textsuperscript{3}, Alan W. Black\textsuperscript{4}, Laurent Besacier\textsuperscript{5}, Sakriani Sakti\textsuperscript{6,7}, Emmanuel Dupoux\textsuperscript{2,8}

\textsuperscript{1}Laboratoire de Linguistique Formelle (CNRS - Paris Diderot - Sorbonne Paris Cité), France
\textsuperscript{2}Cognitive Machine Learning (ENS - CNRS - EHESS - INRIA - PSL Research University), France
\textsuperscript{3}Department of Computer Graphics and Multimedia, Brno Univ. of Technology, Czech Republic
\textsuperscript{4}Language Technologies Institute, Carnegie Mellon University, USA
\textsuperscript{5}Laboratoire d’Informatique de Grenoble, équipe GETALP (Université Grenoble Alpes), France
\textsuperscript{6}Nara Institute of Science and Technology, \textsuperscript{7}RIKEN Center for Advanced Intelligence Project, Japan
\textsuperscript{8}Facebook A.I. Research, Paris, France

*ewan.dunbar@univ-paris-diderot.fr

Abstract

We present the Zero Resource Speech Challenge 2019, which proposes to build a speech synthesizer without any text or phonetic labels: hence, TTS without T (text-to-speech without text). We provide raw audio for a target voice in an unknown language (the Voice dataset), but no alignment, text or labels. Participants must discover subword units in an unsupervised way (using the Unit Discovery dataset) and align them to the voice recordings in a way that works best for the purpose of synthesizing novel utterances from novel speakers, similar to the target speaker’s voice. We describe the metrics used for evaluation, a baseline system consisting of unsupervised subword unit discovery plus a standard TTS system, and a topline TTS using gold phoneme transcriptions. We present an overview of the 19 submitted systems from 11 teams and discuss the main results. Index Terms: zero resource speech technology, speech synthesis, acoustic unit discovery, unsupervised learning

1. Introduction

Young children learn to talk long before they learn to read and write. They can produce novel sentences without being trained on speech annotated with text. Presumably, they achieve this by encoding input speech in their internal phonetic speaker-invariant representations (proto-phonemes), and use this representation to generate speech in their own voice. Reproducing this ability would be useful for the thousands of so-called low-resource languages, which lack the textual resources and/or expertise required to build traditional synthesis systems. The Zero Resource Speech Challenge 2019 (ZR19: \url{www.zerospeech.com/2019/}) proposes to build a speech synthesizer without text or labels. We provide raw audio for the target voice(s) in an unknown language, but no text or labels. Participants must discover subword units in an unsupervised way and align incoming speech to these units in a way that allows for synthesizing novel utterances from novel speakers (see Figure \ref{fig:challenge}). It is a continuation of the sub-word unit discovery track of ZeroSpeech 2017 \cite{dunbar2017zerospeech} and ZeroSpeech 2015 \cite{dunbar2015zerospeech}, as it demands of participants to discover such units, and then evaluate them by assessing their performance on a novel speech synthesis task.

As with the other two challenges, it relies exclusively on freely accessible software and datasets. We provide a baseline system which performs the task using two off-the-shelf components: (a) a system which discovers discrete acoustic units automatically, and (b) a standard TTS system. A submission to the challenge replaces at least one of these systems. The challenge is therefore open to systems which make a contribution primarily to unit discovery, as well as to TTS-only systems which concentrate primarily on improving the quality of the synthesis on the baseline sub-word units. Participants can of course also construct their own end-to-end system with the objective of discovering sub-word units and producing a waveform.

2. Related work

A limited number of papers have provided a proof of concept that TTS without T is feasible \cite{li2018speech, li2019semi}. We use this work as a baseline for the current challenge. This baseline uses out-of-the-box acoustic unit discovery \cite{li2018speech} and an out-of-the-box speech synthesizer (Merlin, with the Ossian front end \cite{liu2019ossian}). Recent improvements on both the unit discovery and the synthesis side of the problem promise to improve on this baseline.

On the acoustic unit discovery side, several methods have been used (binarized autoencoders \cite{aehlert2018binarized}, binarized siamese networks \cite{aehlert2018siamese}: a variety of speaker normalization techniques have been used to improve the categories \cite{aehlert2018binarized}, among other techniques). On the speech synthesis side, waveform generation has recently seen great improvement (Wavenet \cite{kumar2018wavenet}, SampleRNN \cite{kong2018samplernn}, Tacotron 2 \cite{shen2018natural}, DeepVoice3 \cite{li2019deepvoice}, Transformer TTS \cite{shen2018natural}, and others), some of these systems being open source.

Recent research shows that training ASR and TTS jointly with reconstruction losses can result in improvement in both
of the target voice for speech synthesis. The
ment) or one (Surprise), and is for building an acoustic model
would rank such systems separately.
other voice allows us to exclude trivial solutions in which the
new utterances by new speakers. Requiring resynthesis in an-
tune the task of voice conversion.
original audio is returned unchanged.
the pipeline, participants submit the resynthesized audio file; in
the middle of the pipeline, participants give the “pseudo-text”
emboided used at the entry point to the synthesis component.
The general form for the embedding is a sequence of vec-
tors, each one of which can be seen as a “symbol.” The low-
birate constraint (see below) favours a small, finite set of val-
ues for these vectors, as is the case for phoneme units in speech.
The vectors might be one-hot (each “symbol” coded as one on
its own dimension, zero elsewhere), but are not limited to such
representations, and can also be continuous-valued, to take ad-
antage of the similarity structure of the embedding space. To
reduce the bitrate, participants are nevertheless advised to quan-
tize to a discrete subset of values. The number of vectors for a
given test file is not fixed, permitting participants to use a
fixed frame rate, or to instead use “character”-like encodings,
in which successive identical symbols are collapsed and no no-
tion of alignment is retained. The submission format does not
distinguish between these cases. The machine evaluation calcu-
lates the bitrate and the embedding quality; human speakers of
English and Indonesian are presented with the test sentences in
an online experiment in order to evaluate the synthesis quality.

4.1. Synthesis intelligibility, quality and speaker similarity

Intelligibility was measured by asking participants to ortho-
gaphically transcribe the synthesized sentence. Each transcrip-
tion was compared with the gold transcription using the Lev-
enstein distance, yielding a Character Error Rate (CER). The
overall naturalness of the synthesis was assessed on a 1 to 5
scale, yielding a Mean Opinion Score (MOS) [21] Speaker similari-
ty was assessed using a 1 to 5 scale. Sentences were presented in
pairs (target voice, system voice) [21]. A training phase occurred before each
task. Three “catch” trials were included in the trans-
scription, consisting of easy sentences from the original corpus
not included in the rest of the experimental list, allowing us to
detect participants that failed to do the task.
Each participant performed the evaluation tasks in the same
order (Intelligibility, Naturalness, Similarity), the overall eval-
uation lasting about one hour. To avoid re-evaluation of the
same sentence by the same participant, the sentences (types)
were split into two disjoint subsets: one third for the Intelligi-
bility task (62 for English, 49 for Indonesian), and two third for
the Naturalness task (129 for English, 100 for Indonesian). The
complete set of sentences was used in the Similarity task. In
the Intelligibility and Naturalness tasks, all the sentences were
seen by all subjects; in the Similarity task, a pseudo random
one-third of the whole sentences was selected for each partic-
ient. Each sentence token was evaluated at least once with
each system (the submitted, topline and baseline systems, as
well as the original recordings) [22]. English judges were recruited

Additionally, participants could submit two auxiliary embeddings
from earlier or later steps in the systems pipeline in order to analyze the
quality of non-binarized representations computed in the pipeline.

4. Metrics

As shown in Figure 1, participants feed each audio test item to
their system and give two outputs for evaluation: at the end of
the pipeline, participants submit the resynthesized audio file; in

| Dataset       | N speakers | N utt. | Duration |
|---------------|------------|-------|----------|
| Dev: Train Voice | 1 M        | 970   | 2h       |
| Dev: Train Voice | 1 F        | 2563  | 2h40     |
| Dev: Train Unit Disc. | 100       | 5941  | 15h40    |
| Dev: Train Parallel | 10 + 1 M  | 92    | 4.3min   |
| Dev: Test      | 10 + 1 F   | 98    | 4.5min   |
| Sur: Train Voice | 24         | 455   | 28min    |
| Sur: Train Voice | 1 F        | 1862  | 1h30     |
| Sur: Train Unit Disc. | 112       | 15340 | 15h      |
| Sur: Train Parallel | 15 + 1 F  | 150   | 8min     |
| Sur: Test      | 15         | 405   | 29min    |
through Mechanical Turk. Indonesian judges were recruited through universities and research institutes in Indonesia. All were paid the equivalent of 10 USD. Only data from participants with <0.80 CER on catch trials were retained (Development: 35/35; Surprise: 68/69).

4.2. Embedding bitrate and quality

For the bitrate computation, each vector is processed as a character string. A dictionary of the possible values is constructed over the embedding file for the submitted test set. We thus assume that the entire test set corresponds to a sequence of vectors $U$ of length $n$: $U = [s_1, ..., s_n]$. The bit rate for $U$ is then $B(U) = n \sum_{i=1}^{n} p(s_i) \log_2(p(s_i))$, where $p(s_i)$ is the probability of symbol $s_i$. The numerator is $n$ times the entropy of the symbols, which gives the optimal number of bits needed to transmit the sequence of symbols $s_{1..n}$. To obtain a bitrate, we divide by $D$, the total duration of $U$ in seconds.\(^1\)

Since it is unknown whether the discovered representations correspond to particular linguistic units (phone states, phonemes, features, syllables), we evaluate unit quality with a theory-neutral machine ABX score, as in previous Zero Resource challenges.\(^2\) The machine-ABX discriminability between ‘beg’ and ‘bag’ is defined as the probability that $A$ and $X$ are closer than $B$ and $X$, where $A$ and $X$ are tokens of ‘beg’, and $B$ a token of ‘bag’ (or vice versa), and $X$ is uttered by a different speaker than $A$ and $B$. The global ABX discriminability score aggregates over the entire set of minimal pairs such as ‘beg’ ‘bag’ to be found in the test set. The choice of the appropriate distance measure is up to the researcher.

In previous challenges, we used by default the average frame-wise cosine divergence of the representations of the tokens along a DTW-realigned path. We provide in the the option of instead replacing the cosine divergence with the KL divergence, or of using a normalized Levenshtein edit distance over the two sequences. We give ABX scores as error rates (0% for the gold transcription, 50% being chance). Each of the items compared in the ABX task is a triphone ($[izi]-[idi]$, and so on), extracted from the test corpus. Each triphone item is a short chunk of extracted audio, to be decoded by the systems.\(^3\)

The baseline synthesis system we provide requires textual annotations as input, forcing any unit discovery systems using it to convert embeddings into one-hot (unstructured) representations before using them for synthesis. The loss of information incurred may be unwanted. In order to help showcase the unit discovery systems’ performance, we also allow participants to submit up to two auxiliary embeddings, which may be, for example, the outputs of the system prior to quantization. These embeddings are submitted to the ABX and bitrate evaluations, and are represented in light grey in the figures presented below.

120 times in Indonesian, and all possible combinations of sentence and system were seen by at least one participant.\(^4\)

\(^2\) A fixed frame rate transcription may have a higher bitrate than a “textual” representation due to the repetition of symbols across frames. For instance, the bitrate of a 5 ms framewise gold phonetic transcription is around 450 bits/sec and that of a “textual” transcription around 60 bits/sec.

\(^3\) This differs from previous challenges. In previous challenges, longer audio files were provided for decoding, from which the representations of triphones were extracted after the fact using time stamps. In the 2019 edition, triphones are pre-extracted to allow for systems without fixed frame rates. Note that the triphone phone-level language model performs sub-optimally on these files, because they begin with unlikely sequences of phones.

5. Toplines and Baselines

A baseline system is provided, consisting of a pipeline with an acoustic unit discovery system based on DPGMM\(^5\)\(^6\), and a parametric speech synthesizer based on Merlin.\(^7\) As linguistic features, we use contextual information (leading and preceding phones, number of preceding and following phones in current sentence), but no features related to prosody (TOBI), phonic categories (vowel, nasal, and so on) or part-of-speech (noun, verb, adjective, and so on). A topline system is also provided, consisting of an ASR system trained using Kaldi on the original transcriptions. The acoustic model is a tri-state triphone model with 15000 Gaussian mixtures. The language model is a trigram phone-level language model.\(^8\) Output is piped to the TTS system, which is also trained on the gold labels. The baseline system is provided in a container.

6. Results

Nineteen systems were submitted, from eleven groups (see Table 2), using a variety of approaches. Only a few (LI, CH and TJ) used an end-to-end framework. About half used a frame-based encoding (resulting in higher bitrate, but finer temporal information), and half a character-based encoding. Unit discovery methods were diverse (k-means, DPGMM, binarized autoencoders). Most systems used a vocoder, and only two (KA, CH) used direct waveform generation. We present highlights of the results. To access the current leaderboard, including audio samples, see www.zerospeech.com/2019/results.html. In all figures, systems are cored by short names as in Table 2 followed by the submission number from the online leaderboard. Auxiliary embeddings for each submission are followed by A or B.

In previous challenges, the baseline ABX reference for subword embedding quality was calculated on MFCCs, and the topline reference was calculated on supervised posteriorgrams (calculated using DTW with KL-divergence). Here we also provide scores for our unsupervised baseline, and for the topline ASR phone decoding (calculated on Levenshtein distance).

While systems in previous challenges generally improved over the MFCCs, the 2019 challenge was much more difficult, as seen in Figure 2. In the Surprise language, only five final embeddings achieved better performance than the MFCCs (Dev: 25.01%; Surprise: 18.21%). The topline ASR posteriorgram

\(^4\) A word-level language model gives better performance, but we use a phone-level language model in the interest of giving a fair comparison with the subword unit discovery systems asked for in the challenge.

---

**Table 2: Characteristics of the submitted systems**

| System | End-to-end | Frame-based Generation |
|--------|------------|------------------------|
| PA [26] | no | no | Ossian |
| HO [27] | no | yes | ? |
| FE [28] | no | no | Ossian |
| LI [29] | yes | yes | TTSL |
| KA [30] | no | yes | FFTNet |
| KM [Kumar et al.] | no | no | Ossian |
| CH [Cho et al.] | yes | yes | WaveNet |
| RA [Rallabandi et al.] | no | no | Ossian |
| YU [31] | no | no | Ossian |
| GO [Gok et al.] | no | no | Ossian |
| TJ [32] | yes | yes | Inverter |

\(^5\) DPGMM

\(^6\) A fixed frame rate transcription may have a higher bitrate than a “textual” representation due to the repetition of symbols across frames.

\(^7\) A word-level language model gives better performance, but we use a phone-level language model in the interest of giving a fair comparison with the subword unit discovery systems asked for in the challenge.
ABX scores (Dev: 17.22%, Surprise: 8.48%) are comparable to previous challenges. The scores on the ASR decoding are worse than the posteriorgrams (Dev: 29.85%, Surprise: 16.09%)—and worse than the MFCCs, reflecting the fact that the supervised model was fairly simple.

Systems with high ABX error rates have low bitrates, while some systems with relatively high bitrates obtain better scores than the topline (the auxiliary embedding FE-11-A outperforms all others in the Surprise language: among final embeddings, CH-14 shows the best performance, between the decoding and the decoding and posteriorgram topline scores). This suggests that discretizing learned speech embeddings well is hard. It also suggests that relatively dense representations, in spite of containing strictly more information than the phonemic transcriptions, are still useful in settings where only the linguistically relevant contrasts are necessary. The “least discrete” representations have a bitrate of 0.19, max 0.28. Differences between systems should be interpreted in light of these confidence intervals.

0.19, max 0.28. Differences between systems should be interpreted in light of these confidence intervals.

Note, however, that Similarity only measures closeness to the target voice: low similarity does not distinguish between mere reproduction of the source voice, versus synthesis close to neither target nor source.

The submitted non-discrete auxiliary embeddings showed generally better ABX scores than their discrete counterpart, at the expense of a higher bitrate, some of them even beating the supervised posteriors (FE: Dev: 13.82%, Surprise: 6.52%).

We also calculate bootstrap (N=10000) 95% confidence intervals for each of these measures, for each system. We calculated half the width of the CI for each submission, in each of the two languages. We report the mean and the max (worst-case) CI half-width. For the Development language: CER, mean 0.04, max 0.07; MOS, mean 0.13, max 0.17; Similarity, mean 0.24, 0.28. For the Surprise language: CER, mean 0.03, max 0.05; MOS, mean 0.10, max 0.14; Similarity, mean 0.19, max 0.28. Differences between systems should be interpreted in light of these confidence intervals.

Systems with high ABX error rates have low bitrates, while some systems with relatively high bitrates obtain better scores than the topline (the auxiliary embedding FE-11-A outperforms all others in the Surprise language: among final embeddings, CH-14 shows the best performance, between the decoding and the decoding and posteriorgram topline scores). This suggests that discretizing learned speech embeddings well is hard. It also suggests that relatively dense representations, in spite of containing strictly more information than the phonemic transcriptions, are still useful in settings where only the linguistically relevant contrasts are necessary. The “least discrete” representations have a bitrate of 0.19, max 0.28. Differences between systems should be interpreted in light of these confidence intervals.

Note, however, that Similarity only measures closeness to the target voice: low similarity does not distinguish between mere reproduction of the source voice, versus synthesis close to neither target nor source.

The submitted non-discrete auxiliary embeddings showed generally better ABX scores than their discrete counterpart, at the expense of a higher bitrate, some of them even beating the supervised posteriors (FE: Dev: 13.82%, Surprise: 6.52%).

We also calculate bootstrap (N=10000) 95% confidence intervals for each of these measures, for each system. We calculated half the width of the CI for each submission, in each of the two languages. We report the mean and the max (worst-case) CI half-width. For the Development language: CER, mean 0.04, max 0.07; MOS, mean 0.13, max 0.17; Similarity, mean 0.24, 0.28. For the Surprise language: CER, mean 0.03, max 0.05; MOS, mean 0.10, max 0.14; Similarity, mean 0.19, max 0.28. Differences between systems should be interpreted in light of these confidence intervals.
9. References

[1] M. Versteegh, X. Anguera, A. Jansen, and E. Dupoux, “The Zero Resource Speech Challenge 2015: Proposed approaches and results,” Procedia Computer Science, vol. 81, pp. 67–72, 2016.

[2] E. Dunbar, X. N. Cao, J. Benjumea, J. Karalayas, M. Bernard, L. Besacier, X. Anguera, and E. Dupoux, “The Zero Resource Speech Challenge 2017,” in ASRU 2017. IEEE, 2017, pp. 323–330.

[3] P. K. Muthukumar and A. W. Black, “Automatic discovery of a phonetic inventory for unwritten languages for statistical speech synthesis,” in IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2014, Florence, Italy, May 4–9, 2014, pp. 2594–2598.

[4] O. Scharenborg, L. Besacier, A. W. Black, M. Hasegawa-Johnson, F. Metze, G. Neubig, S. Stüker, P. Godard, M. Müller, L. Ondel, S. Palaskar, P. Arthur, F. Ciannella, M. Du, E. Larsen, D. Merkx, R. Riad, L. Wang, and E. Dupoux, “Linguistic unit discovery from multi-modal inputs in unwritten languages. Summary of the "speaking rosetta" JSALT 2017 workshop,” in ICASSP. IEEE, 2018, pp. 4979–4983.

[5] L. Ondel, L. Burget, and J. Cernocky, “Variational inference for acoustic unit discovery,” in SLTU, ser. Procedia Computer Science, vol. 81. Elsevier, 2016, pp. 80–86.

[6] Z. Wu, O. Watts, and S. King, “Merlin: An open source neural network speech synthesis system,” in Speech Synthesis Workshop, ISCA, 2016, pp. 202–207.

[7] L. Badjatia, C. Canevari, L. Fudiga, and G. Metta, “An auto-encoder based approach to unsupervised learning of subword units,” in ICASSP. IEEE, 2014, pp. 7634–7638.

[8] A.-F. Myrman and G. Salvi, “Partitioning of posteriorgrams using siamese models for unsupervised acoustic modelling,” in International Workshop on Grounding Language Understanding (GLU). ISCA, 2017.

[9] M. Heck, S. Sakti, and S. Nakamura, “Feature optimized DPGM clustering for unsupervised subword modeling: A contribution to ZeroSpeech 2017,” in ASRU 2017. IEEE, 2017. pp. 740–747.

[10] A. van den Oord, D. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. W. Senior, and K. Kavukcuoglu, “Wavenet: A generative model for raw audio,” in 3SW. ISCA, 2016, p. 125.

[11] S. Mehri, K. Kumar, I. Guirajani, R. Kumar, S. Jain, J. Sotelo, A. C. Courville, and Y. Bengio, “SampleRNN: An unconditional end-to-end neural audio generation model,” CoRR, vol. abs/1612.07837, 2016.

[12] J. Shen, R. Pang, R. J. Weiss, M. Schuster, N. Jaitly, Z. Yang, Z. Chen, Y. Zhang, Y. Wang, R. Ryan, A. R. Saurous, Y. Agiomyrgianakis, and Y. Wu, “Natural TTS synthesis by conditioning wavenet on MEL spectrogram predictions,” in ICASSP. IEEE, 2018, pp. 4779–4783.

[13] W. Ping, K. Peng, A. Giblesky, S. Ö. Arik, A. Kannan, S. Narang, J. Raiman, and J. Miller, “Deep voice 3: 2000-speaker neural text-to-speech,” CoRR, vol. abs/1710.07654, 2017.

[14] N. Li, S. Liu, Y. Liu, S. Zhao, M. Liu, and M. Zhou, “Close to human quality TTS with transformer,” CoRR, vol. abs/1809.08895, 2018.

[15] A. Tjandra, S. Sakti, and S. Nakamura, “Listening while speaking: Speech chain by deep learning,” in ASRU 2017, 2017, pp. 301–308.

[16] T. Kaneko and H. Kameoka, “Parallel-data-free voice conversion using cycle-consistent adversarial networks,” CoRR, vol. abs/1711.11293, 2017.

[17] C. Hsu, H. Hwang, Y. Wu, Y. Tsao, and H. Wang, “Voice conversion from non-parallel corpora using variational auto-encoder,” in Asia-Pacific Signal and Information Processing Association Annual Summit and Conference. APSIPA 2016, Jeju, South Korea, December 13-16, 2016, 2016, pp. 1–6.

[18] J. Chou, C. Yeh, H. Lee, and L. Lee, “Multi-target voice conversion without parallel data by adversarially learning disentangled audio representations,” CoRR, vol. abs/1804.02812, 2018.

[19] Y. Gao, R. Singh, and B. Raj, “Voice impersonation using generative adversarial networks,” in ICASSP. IEEE, 2018, pp. 2506–2510.

[20] A. van den Oord, O. Vinyals et al., “Neural discrete representation learning,” in Advances in Neural Information Processing Systems, 2017, pp. 6306–6315.

[21] J. Chorowski, R. Weiss, S. Bengio, and A. v. d. Oord, “Unsupervised speech representation learning using wavenet autoencoders,” arXiv preprint arXiv:1901.08910, 2019.

[22] S. Sakti, R. Maia, S. Sakai, T. Shimizu, and S. Nakamura, “Development of HMM-based Indonesian speech synthesis,” in Proc. Oriental COCOSDA, 2008, pp. 215–219.

[23] 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.

[24] L. Ondel, P. Godard, L. Besacier, E. Larsen, M. Hasegawa-Johnson, O. Scharenborg, E. Dupoux, L. Burget, F. Yvon, and S. Khudanpur, “Bayesian models for unit discovery on a very low resource language,” in 2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2018, pp. 5939–5943.

[25] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Gymrek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz et al., “The Kaldi speech recognition toolkit,” IEEE Signal Processing Society, Tech. Rep., 2011.

[26] K. Pandia and H. Murthy, “Zero resource speech synthesis using transcriptions derived from perceptual acoustic units,” Submitted to INTERSPEECH 2019, 2019.

[27] Y. Bin and W. Yuan, “A VAE model with speaker verification for unsupervised subword modeling: A submission to ZeroSpeech 2019,” Submitted to INTERSPEECH 2019, 2019.

[28] S. Feng, T. Lee, and Z. Peng, “Combining adversarial training and disentangled speech representation for robust zero-resource subword modeling,” Submitted to INTERSPEECH 2019, 2019.

[29] A. Liu, P.-C. Hsu, and H.-Y. Lee, “Unsupervised end-to-end learning of discrete linguistic units for voice conversion,” Submitted to INTERSPEECH 2019, 2019.

[30] R. Elloff, A. Nortje, B. L. van Niekerk, A. Govender, L. Nortje, A. Pretorius, E. van Biljon, E. van der Westhuizen, L. van Staden, and H. Kamer, “Unsupervised acoustic unit discovery for speech synthesis using discrete latent-variable neural networks,” Submitted to INTERSPEECH 2019, 2019.

[31] B. Yusuf, A. Gok, B. Gundogdu, O. D. Kose, and M. Saracel, “Temporally-Aware Acoustic Unit Discovery for ZeroSpeech 2019 Challenge,” Submitted to INTERSPEECH 2019.

[32] 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.”