INTERSPEECH 2019
September 15–19, 2019, Graz, Austria

Temporally-Aware Acoustic Unit Discovery for Zerospeech 2019 Challenge

Bolaji Yusuf[1], *Alican Gök*[1], *Batuhan Gundogdu*[1,2], *Oyku Deniz Kose*[1], *Murat Saraclar*[1]

[1]Bogazici University, Turkey
[2]National Defense University Naval Academy, Turkey
{bolaji.yusuf,alican.gok,batuhan.gundogdu,deniz.kose,murat.saraclar}@boun.edu.tr

Abstract

Zero-resource speech processing efforts focus on unsupervised discovery of sub-word acoustic units. Common approaches work with spatial similarities between the acoustic frame representations within Bayesian or neural network–based frameworks. We propose two methods that utilize the temporal proximity information in addition to the acoustic similarity for clustering frames into acoustic units. The first approach uses a temporally biased self-organizing map (SOM) to discover such units. Since the SOM unit indices are correlated with (vector) spatial distance, we pool neighboring units and then train a recurrent neural network to predict each pooled unit. The second approach incorporates temporal awareness by training a recurrent sparse autoencoder, in which unsupervised clustering is done on the intermediate softmax layer. This network is then fine-tuned using aligned pairs of acoustically similar sequences obtained via unsupervised term discovery. Our approaches outperform the provided baseline system on two main metrics of the Zerospeech 2019 challenge, ABX-discriminability and bitrate of the quantized embeddings, both for English and the surprise language. Furthermore, the temporal-awareness and the post-filtering techniques adopted in this work resulted in an enhanced continuity of the decoding, yielding low bitrates.

Index Terms: self-organizing map, gated recurrent unit, sparse recurrent autoencoder, correspondence autoencoder

1. Introduction

Current speech technologies rely on large, extensively labeled corpora to achieve good performance. Alas, such corpora are only available in a small portion of languages due to the cost of annotation. This is further compounded for languages with a small population of native speakers or languages without an orthographic form. Therefore, there have been a few efforts toward bridging the gap between speech processing for resource-rich languages and the so-called low-resource languages, including the IARPA Babel program [1], the MediaEval spoken web search task [2], and the Zerospeech challenges [3].

Previous iterations of the Zerospeech challenge have focused on discovery of acoustic units from a spoken training corpus without any annotations. These learned representations are required to allow utterance discriminability in vector space (e.g. with DTW). The Zerospeech 2019 [4] extends this task by also including a synthesis component. In addition to discriminability, knowledge of the learned representation must be enough to re-synthesize the utterance. The task is thus:

- Train units: Using the provided training corpus, learn a set of representations corresponding to each acoustic unit.
- Train voice: For a set of target speakers, decode each target speaker's training utterances into the learned units, and use this alignment information to train a speech synthesis system for that speaker.
- Test: Decode the test utterance and use the decoded sequence to synthesize the same utterance in the target speaker's voice.

In the context of acoustic unit discovery, one approach has been to use infinite mixture Bayesian models with Dirichlet-process priors such as Dirichlet-process HMM-GMMs [5, 6] and Dirichlet-process GMMs [7, 8]. Another approach has been to use neural networks in the form of autoencoders [9, 10] or Siamese-style networks with labels obtained from unsupervised term discovery (UTD) systems [11, 12], temporal proximity [13] or GMM clustering [14].

In the first of our two submissions, we train a self organizing map [15] to cluster the speech frames into ordered classes, and a gated recurrent unit (GRU) neural network to predict these units. We modify the training of the SOM to incorporate neighboring frames when updating a class to account for the temporal continuity of the acoustic frames. Since unit index distance also corresponds to spatial distance, we pool SOM classes within some neighborhood to model within-class variability and form acoustic units which the GRU is then trained to predict. Each utterance is then represented as a sequence of one-hot vectors obtained by quantizing (hard maximum) the output of the GRU's softmax layer and removing repetitions.

For the second submission, we adopt a recurrent (GRU) autoencoder with a hidden softmax layer whose dimensions correspond to the acoustic units. The network is first trained to minimize the reconstruction error on the input sequence with an additional sparsity cost (to reduce quantization error) on the intermediate representation. Furthermore, we use a UTD [16] to detect matching sequences and use DTW to align them at the frame level. The recurrent autoencoder is then finetuned (with a lower learning rate) to construct corresponding frames from each other. The final representation is obtained by quantizing the intermediate representation and removing repetitions.

For the synthesis, we use the provided baseline system, which is based on the Merlin toolkit [17] in both submissions. It was necessary to quantize our representations since the synthesizer requires one-hot embeddings with no unit repetitions. The next section describes our systems in more detail and in Section 3, empirical results are provided.

2. System Description

We have two systems which are described below. The first is a Self-Organizing Map-Recurrent Neural Network (SOM-RNN) hybrid, and the second is a Correspondence Recurrent Sparse Autoencoder (CoRSA).

http://dx.doi.org/10.21437/Interspeech.2019-1430

2.1. SOM-RNN

For the first system, we use an SOM to cluster the frames and train an RNN to predict the (subsampled) cluster labels. The SOM provides a way to obtain labels for the RNN training. Moreover, the correlation between neighboring frames of the SOM allow us to model within-class variability.

2.1.1. *SOM with temporal continuity*

SOMs perform an iterative unsupervised clustering of vectors into ordered units such that neighboring units have a low distance in vector space. We modify the SOM in this work to also cluster temporally proximal frames into neighboring units in order to leverage the temporal continuity inherent in speech.

Given a sequence of speech frames, $\mathcal{X} = [\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N]$, the task is to assign the each frame to one of a set of centroid vectors, $\mathcal{U} = \{\mathbf{u}^1, \mathbf{u}^2, \ldots, \mathbf{u}^c\}$. For training, first we define the exponential window function:

$$w_\alpha(p, q) = e^{-\alpha(p-q)^2}. \tag{1}$$

For each $\mathbf{x}_t \in \mathcal{X}$, we determine the "winning" class:

$$d_t = \text{argmin}_i ||\mathbf{u}^i - \bar{\mathbf{x}}_t||^2 \tag{2}$$

and update the units in its neighborhood:

$$\mathbf{u}^i(m+1) = \mathbf{u}^i(m) + \eta(m) \cdot w_{\alpha_u}(d_t, i)(\bar{\mathbf{x}}_t - \mathbf{u}^i), \tag{3}$$

where m is the iteration index and $\bar{\mathbf{x}}_t$ is a weighted average of the frames around $\mathbf{x}_t$:

$$\bar{\mathbf{x}}_t = \frac{\sum_{n=1}^{N} w_{\alpha_t}(t, n)\mathbf{x}_n}{\sum_{n=1}^{N} w_{\alpha_t}(t, n)}. \tag{4}$$

Thus, each unit is updated not just with a single frame but with a succession of them. Note that α_u and α_t are hyperparameters that control the rate of decay of the unit and temporal windows respectively, while $\eta(m)$ is the learning rate.

The units are initialized by randomly sampling from the training features. The sampling procedure is repeated several times and the permutation with the most variance is kept. Then the training procedure is iterated until convergence. Afterwards, a sequence of labels $D = [d_1, d_2, \ldots, d_N]$ is obtained and used for RNN training.

2.1.2. *RNN with SOM labels*

After training the SOM, we train an RNN to predict the labels obtained from the SOM units. However, since neighboring units in the SOM are correlated, we pool them together to obtain a new sequence of labels, $\tilde{D} = [\tilde{d}_1, \tilde{d}_2, \ldots, \tilde{d}_N]$ such that:

$$\tilde{d}_i = \lceil d_i/k \rceil \tag{5}$$

where k is the pooling width of each unit. Thus instead of using each unit, we use a collection of neighboring units to represent a frame; this pooling can be thought of as mapping from context-dependent to context-independent phones, and we found that it improves the ABX by about 0.03 on the development set.

We train a bidirectional GRU with 2 hidden layers with 24 units and a dropout rate of 0.4 on each layer to predict the new labels. The optimization is done with Adam [18] using the default parameters, including a learning rate of 10^{-3}. The other hyperparameters of the system are given in Table 1.

Table 1: *SOM-RNN Hyperparameters*

Parameter	c	α_t	α_u	k	$\eta(0)$
Value	128	0.5	0.1	4	0.01

2.1.3. *Embeddings description*

In each submission, we provide three representations on which our systems are to be evaluated. For the SOM-RNN system, these are:

(i) Test: This is the final representation on which synthesis is performed. It is obtained from the softmax output of the GRU. Since the baseline synthesis system which we use requires embeddings in one-hot form with no repetitions, we quantize the softmax outputs thus.

(ii) Auxiliary-1: This representation is obtained from the SOM directly. For a given frame, its distance from each cluster centroid is computed. The distances are multiplied by 0.01 (a hyperparameter) and their softmax is computed get the embedding.

(iii) Auxiliary-2: This embedding is the same as the Test embedding, except that two of the hyperparemeters are changed, namely $c = 64$ and $k = 3$. This results in a trade-off some ABX discriminability for reduced bitrate.

2.2. Correspondence Recurrent Sparse Autoencoder

In the second system, we trained a recurrent sparse autoencoder (RSA) to obtain an alternative and compressible frame-level representation. The decoder (and the encoder) contain a gated recurrent unit (GRU) based RNN layer and a feed-forward dense layer. Speech features are fed into the input layer of this autoencoder and a sparse representation is obtained through an intermediate feed-forward softmax layer, using the hidden state activations of the network at each timestep, yielding a posteriorgram-like sequence. This intermediate representation is expected to act as the 'hidden' or the 'unknown' states (the sub-word units to be discovered) of the sequence observation. To enforce sparsity of this representation, this layer is penalized with negative L2-norm, so that the representation would become closer to one-hot vectors, hence reducing the quantization error when they are eventually converted to one-hot vectors for compression and baseline synthesizer training.

In addition to the sparsity, a temporal continuity cost is also applied to the intermediate layer so that the consecutive frames are more likely to *soft-output* the same state. The RSA architecture is given in Figure 1 (left). The input sequence $\mathbf{x}_t \in \mathcal{R}^{D \times T}$ produces the sparse posteriorgram representation $\mathbf{p}_t \in [0, 1]^{K \times T}$ via a GRU layer and a feed forward softmax layer, which is then used to generate $\hat{\mathbf{x}}_t$ at the output, to emulate $\mathbf{x}_t$, where D and K are the input dimensionality and the number of states, respectively. The combined cost is given in (6).

$$J = \sum_{t \in 1 \cdots T} ||\mathbf{x}_t - \hat{\mathbf{x}}_t||^2 - \lambda_1 ||\mathbf{p}_t||_2^2 + \lambda_2 ||\mathbf{p}_t - \mathbf{p}_{t-1}||^2 \tag{6}$$

This network is then fine tuned in similar fashion to the correspondence autoencoder [19], with pairs of acoustic segments belonging to the same cluster. These pairs are obtained using an unsupervised term discovery (UTD) system [16], and the frames of each pair are aligned through dynamic time warping

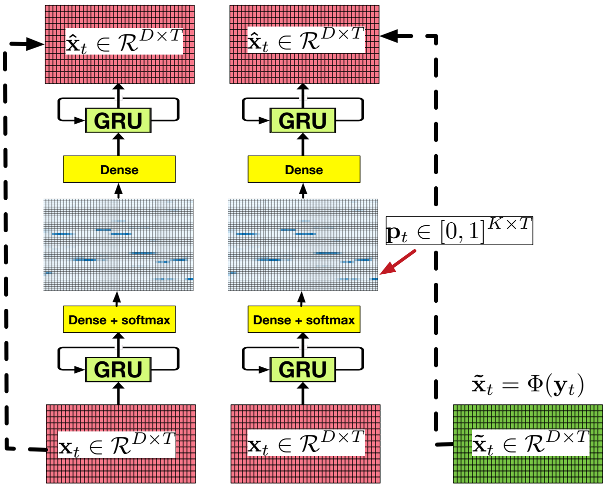
$\hat{\mathbf{x}}_t \in \mathcal{R}^{D\times T}$
$\hat{\mathbf{x}}_t \in \mathcal{R}^{D\times T}$
GRU
GRU
Dense
Dense
$\mathbf{p}_t \in [0,1]^{K\times T}$
Dense + softmax
Dense + softmax
GRU
GRU
$\tilde{\mathbf{x}}_t = \Phi(\mathbf{y}_t)$
$\mathbf{x}_t \in \mathcal{R}^{D\times T}$
$\mathbf{x}_t \in \mathcal{R}^{D\times T}$
$\tilde{\mathbf{x}}_t \in \mathcal{R}^{D\times T}$

Figure 1: *The RSA (left) and the correspondence RSA (right)*

Table 2: *CoRSA Hyperparameters*

Parameter	Value
# GRU neurons	128
K (embedding dimension)	64
T (for RSA training)	150
λ_1	0.2
λ_2	0

(DTW). The correspondence autoencoder (cAE) system presented in [19] is frame-level, i.e. the frames that are believed to belong to the same class are used to generate each other in through a feed-forward autoencoder. The cAE in [20] contains RNNs, but the main idea there is to obtain a fixed length embedding which is used to generate the similar sequences. In this work, however, we align the similar sequences using DTW so that a combination of [19] and [20] is obtained to obtain a better representation of a sparse posteriorgram on the intermediate layer. The correspondence RSA (CoRSA) is depicted in Figure 1. All pairs of similar sequences $\mathbf{x}_t$ and $\mathbf{y}_t$, obtained via UTD are aligned via DTW to yield the alignment path Φ. The RSA network is then trained to generate the aligned sequence $\tilde{\mathbf{x}}_t = \Phi(\mathbf{y}_t)$ from the encoding obtained by feeding $\mathbf{x}_t$ on the input. It should be noted the reverse is also carried out, that is $\tilde{\mathbf{y}}_t = \Phi^{-1}(\mathbf{x}_t)$ is also used with $\mathbf{y}_t$ as input.

The network is trained with Adam optimization [18], with a learning rate of 10^{-3} for the RSA, and 10^{-4} for fine tuning it via CoRSA. The other hyperparameters of the system are given in Table 2. Note that we decided to use $\lambda_2 = 0$ in the final submission as it led to better results on the discriminability task, probably at a cost of reduced synthesis accuracy. We have also experimented with stacked GRU layers at the decoder and encoder. As expected, these networks did a better job at reconstructing the speech features, however the learned intermediate representations turned out to perform significantly worse on the discriminability task.

2.2.1. *Temporal Filtering*

The low bit-rate embeddings of the test utterances are obtained by treating the posteriorgram as the sequence of posterior probabilities of the underlying sub-word units. Despite the recurrent structure of the encoder and the continuity cost included in (6), the one-hot representation exhibits erroneous and unmeaningfully short state assignments on the encoding. To alleviate the adverse effects of such behavior on both discriminability and synthesis tasks, we applied a temporal continuity filtering on the one-hot sequence $\mathbf{o}_t$, obtained by detecting the maximum activation of each frame of $\mathbf{p}_t$ by taking the median of each state activation on a sliding window. Figure 2 demonstrates the embedding obtained by CoRSA and the effect of temporal filtering, on one sample real utterance of Zerospeech 2019 English test set. The posteriorgram on the left is very much like a phone posteriorgram or a Gaussian posteriorgram, even though no supervision was applied on this representation during training. The median filter corrects the islands of possibly incorrect state assignments and reinforces long lasting sequences. This procedure helped notably in reducing the bit-rate without drastically hurting the discriminability test performance.

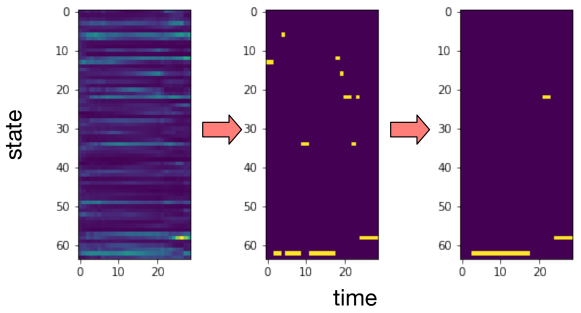

Figure 2: *Post filtering for efficient temporal compression, the posteriorgram is reduced into a* 3 *symbol sequence*

2.2.2. *Embeddings description*

The three representations generated by this system in our submission to the Zerospeech 2019 challenge are as follows:

(i) Test: The maximally-compressed embedding, obtained by removing the repeating sequences of the median-filtered $\mathbf{o}_t$. The filter order is taken to be 5. For instance, the sample test sequence in Figure 2 is compressed down to 3 states.

(ii) Auxiliary-1: Frame-level sparse intermediate posteriorgram activations converted to one-hot vectors ($\mathbf{o}_t$)

(iii) Auxiliary-2: Frame-level sparse intermediate posteriorgram activations ($\mathbf{p}_t$), no quantization is applied.

3. Experiments

3.1. Experiment Setup and Data Description

The experiments are conducted on the two languages provided in the challenge: English and the surprise language. The English language corpus is used for system development and hyperparameter selection, while the surprise language corpus, later revealed to be Standard Indonesian collected in [21, 22], is used strictly for test. Within each language, there are three data sets of interest:

(i) Unit training set: A collection of utterances (about 15h total per language) by numerous speakers to be used for training the acoustic unit discovery model.

(ii) Voice training set: A collection of utterances by the target speakers to be used for training acoustic models for synthesizing each speaker's speech. The development language has two target speakers, each with over 2 hours of training utterances, while the surprise language has one target speaker with about an 90 minutes of training data.

(iii) Test set: A collection of utterances (about 30 minutes total per language) which are to be decoded into the discovered units. Some of these utterances are also to be used for synthesis in the target speakers' voices.

The input features for both our systems are 13-dimensional PLP features. We perform per-speaker cepstral mean and variance normalization (CMVN) and a further pre-filtering (exponential decay with rate 0.95 across feature dimensions) on the PLP features to reduce the effects of speaker variability.

3.2. Evaluation Metrics

The submissions to the Zerospeech 2019 challenge have been evaluated based on the discriminability performance and the bitrate of their embeddings, and the quality of the synthesized wave files using these embeddings.

To assess the discriminability of the produced embeddings, a machine ABX score, also referred to as ABX error rate, is calculated as in the previous Zero Resource challenges [23]. The synthesized wave files are evaluated by the judges based on their intelligibility through the transcription character error rates, speaker similarity, and the overall quality of the synthesis on a 1 to 5 scale, yielding a Mean Opinion Score (MOS).

3.3. Experimental Results

The provided baseline system consists of an acoustic unit discovery system based on HMM [6] with Dirichlet process priors, and a speech synthesizer based on Merlin [17], trained in an unsupervised fashion using the same available data to the participants. The topline system on the other hand, is trained using supervision (on gold labels), and consists of a pipeline of an automatic speech recognition system piped to a trained TTS system based on Merlin.

The performance of our proposed methods compared to the baseline and topline for the ABX and MOS metrics are listed in Table 3. Our systems' results along with other submissions to the challenge are provided in Figures 3 and 4.

For the surprise language, the test embedding of CoRSA and the second auxiliary embedding of SOM-RNN outperform the baseline system in terms of both ABX and bitrate, thus achieving a more compressed and discriminative representation of the speech. The temporal-awareness and post-filtering techniques adopted in CoRSA especially resulted in enhanced continuity of the embeddings, and achieved a higher level of compression than even the topline system. Although the SOM-RNN outperforms the topline on the ABX task in the development language, it does not do so in the surprise language implying a sensitivity to hyperparameter choice.

Since our contribution was purely to the unit discovery part of the challenge, we used the baseline synthesis system, and our synthesis results were on par with the general trend of the MOS vs. bitrate performances.

Table 3: *The performance of the proposed systems on Indonesian and English. X is the SOM-RNN system and Y is the CoRSA system as described in the previous section.*

	Surprise (Indonesian)			**English**		
System	**MOS**	**ABX**	**Bitrate**	**MOS**	**ABX**	**Bitrate**
Baseline	2.07	27.46	74.55	2.5	35.63	71.98
Topline	3.92	16.09	35.20	2.77	29.85	37.73
X	1.84	24.16	121.03	2.42	25.69	92.37
X - aux1	-	12.29	1732.67	-	18.92	1216.31
X - aux2	-	24.46	66.86	-	28.37	49.97
Y	1.46	27.26	29.46	2.02	35.86	34.66
Y - aux1	-	22.21	279.48	-	30.35	275.85
Y - aux2	-	10.87	1732.67	-	18.57	1216.31

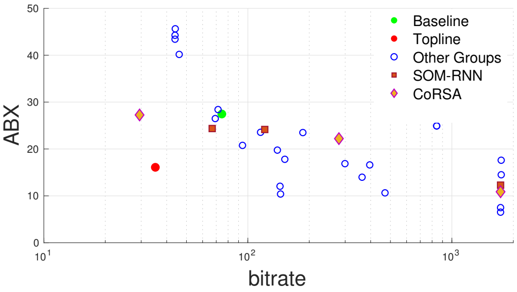

Figure 3: *ABX scores vs. bitrate for all embeddings produced by the challenge participants on the surprise language*

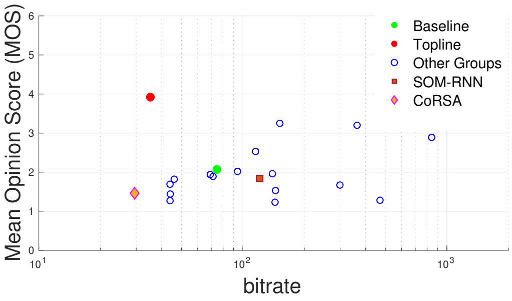

Figure 4: *MOS vs. bitrate for the synthesized wave files of the challenge participants on the surprise language*

4. Conclusion

We have described the two systems our group submitted to the ZeroSpeech 2019 challenge, primarily contributing to the unit-discovery aspect. The first approach used a self organizing map to cluster the speech frames into classes, and an RNN to predict these units, whereas the second approach utilized a novel RNN-based sparse autoencoder and temporal post-filtering. Although we were unable to improve on the synthesis part of the challenge, both of systems performed better than the provided baseline system on the ABX discrimination task.

5. Acknowledgements

This study was supported in part by the Scientific and Technological Research Council of Turkey (TUBITAK) under Project 116E076, and BIDEB 2211.

6. References

[1] M. Harper, *IARPA Babel program*, 2014, accessed at June 2018. [Online]. Available: https://www.iarpa.gov/index. php/research-programs/babel

[2] F. Metze, X. Anguera, E. Barnard, M. Davel, and G. Gravier, "The spoken web search task at mediaeval 2012," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*. IEEE, 2013, pp. 8121–8125.

[3] M. Versteegh, R. Thiolliere, T. Schatz, X. N. Cao, X. Anguera, A. Jansen, and E. Dupoux, "The zero resource speech challenge 2015," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[4] 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 *the 20th Annual Conference of the International Speech Communication Association (INTERSPEECH 2019): Crossroads of Speech and Language*, 2019.

[5] C. Y. Lee and J. Glass, "A nonparametric bayesian approach to acoustic model discovery," in *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics: Long Papers-Volume 1*. Association for Computational Linguistics, 2012, pp. 40–49.

[6] L. Ondel, L. Burget, and J. Černockỳ, "Variational inference for acoustic unit discovery," *Procedia Computer Science*, vol. 81, pp. 80–86, 2016.

[7] H. Chen, C.-C. Leung, L. Xie, B. Ma, and H. Li, "Parallel inference of dirichlet process gaussian mixture models for unsupervised acoustic modeling: A feasibility study," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[8] M. Heck, S. Sakti, and S. Nakamura, "Unsupervised linear discriminant analysis for supporting dpgmm clustering in the zero resource scenario," *Procedia Computer Science*, vol. 81, pp. 73–79, 2016.

[9] L. Badino, A. Mereta, and L. Rosasco, "Discovering discrete subword units with binarized autoencoders and hidden-markov-model encoders," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[10] H. Kamper, M. Elsner, A. Jansen, and S. Goldwater, "Unsupervised neural network based feature extraction using weak top-down constraints," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 5818–5822.

[11] R. Thiolliere, E. Dunbar, G. Synnaeve, M. Versteegh, and E. Dupoux, "A hybrid dynamic time warping-deep neural network architecture for unsupervised acoustic modeling," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[12] N. Zeghidour, G. Synnaeve, M. Versteegh, and E. Dupoux, "A deep scattering spectrum—deep siamese network pipeline for unsupervised acoustic modeling," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 4965–4969.

[13] G. Synnaeve and E. Dupoux, "A temporal coherence loss function for learning unsupervised acoustic embeddings," *Procedia Computer Science*, vol. 81, pp. 95–100, 2016.

[14] A. Fahlström Myrman and G. Salvi, "Partitioning of posteriorgrams using siamese models for unsupervised acoustic modelling," in *Proc. GLU 2017 International Workshop on Grounding Language Understanding*, 2017, pp. 27–31. [Online]. Available: http://dx.doi.org/10.21437/GLU.2017-6

[15] T. Kohonen, *Self-organizing maps*. Springer Science & Business Media, 2012, vol. 30.

[16] A. Jansen and B. Van Durme, "Efficient spoken term discovery using randomized algorithms," in *2011 IEEE Workshop on Automatic Speech Recognition & Understanding*. IEEE, 2011, pp. 401–406.

[17] 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. [Online]. Available: http://dx.doi.org/10.21437/SSW.2016-33

[18] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[19] D. Renshaw, H. Kamper, A. Jansen, and S. Goldwater, "A comparison of neural network methods for unsupervised representation learning on the zero resource speech challenge," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[20] H. Kamper, "Truly unsupervised acoustic word embeddings using weak top-down constraints in encoder-decoder models," *arXiv preprint arXiv:1811.00403*, 2018.

[21] 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.

[22] 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.

[23] T. Schatz, V. Peddinti, F. Bach, A. Jansen, H. Hermansky, and E. Dupoux, "Evaluating speech features with the minimal-pair ABX task: Analysis of the classical MFC/PLP pipeline," in *INTERSPEECH 2013: 14th Annual Conference of the International Speech Communication Association*, 2013, pp. 1–5.