Evaluating the reliability of acoustic speech embeddings

Robin Algayres, Mohamed Zaiem, Benoît Sagot, Emmanuel Dupoux

Evaluating the reliability of acoustic speech embeddings

Robin Algayres[1,2]*, Mohamed Salah Zaiem*[1,2]*, Benoît Sagot*[2]*, Emmanuel Dupoux*[1,2]

[1]ENS-PSL, EHESS, CNRS, Paris [2]Inria, Paris

robin.algayres@inria.fr, mohamed.zaiem@inria.fr, benoit.sagot@inria.fr, emmanuel.dupoux@inria.fr

Abstract

Speech embeddings are fixed-size acoustic representations of variable-length speech sequences. They are increasingly used for a variety of tasks ranging from information retrieval to unsupervised term discovery and speech segmentation. However, there is currently no clear methodology to compare or optimize the quality of these embeddings in a task-neutral way. Here, we systematically compare two popular metrics, ABX discrimination and Mean Average Precision (MAP), on 5 languages across 17 embedding methods, ranging from supervised to fully unsupervised, and using different loss functions (autoencoders, correspondance autoencoders, siamese). Then we use the ABX and MAP to predict performances on a new downstream task: the unsupervised estimation of the frequencies of speech segments in a given corpus. We find that overall, ABX and MAP correlate with one another and with frequency estimation. However, substantial discrepancies appear in the fine-grained distinctions across languages and/or embedding methods. This makes it unrealistic at present to propose a task-independent silver bullet method for computing the intrinsic quality of speech embeddings. There is a need for more detailed analysis of the metrics currently used to evaluate such embeddings.

Index Terms: unsupervised speech processing, speech embeddings, frequency estimation, evaluation metrics, representation learning, k-nearest neighbours

1. Introduction

Unsupervised representation learning is the area of research that aims to extract units from unlabelled speech that are consistent with the phonemic transcription [1–4]. As opposed to text, speech is subject to large variability. Two speech sequences with the same transcription can have significantly different raw speech signals. In order to work on speech sequences in an unsupervised way, there is a need for robust acoustic representations. To address that challenge, recent methods use *speech embeddings*, i.e. fixed-size representations of variable-length speech sequences [5–12].[1]

Speech embeddings can be used in many applications, such as key-word spotting [13–15], spoken term discovery [16–18], and segmentation of speech into words [19–21]. It is convenient to evaluate the reliability of speech embeddings without being tied to a particular downstream task. One way to do that is to compute the intrinsic quality of speech embeddings. The basic idea is that a reliable speech embedding should maximize the information relevant to its type and minimize irrelevant token-specific information. Two popular metrics have been used: the mean average precision (MAP) [22] and the ABX discrimination score [23].

[1]A speech sequence is a non-silent part of the speech signal (not necessarily a word). It can be transcribed into a phoneme n-gram.

ABX and MAP are mathematically distinct yet they are expected to correlate well with each other as they both evaluate the discriminability of speech embeddings in terms of their transcription. However, [6] revealed a surprising result: the best model according to the ABX, is also the worst one according to the MAP. Following [6]'s results, we observed that this kind of discrepancies is much more common than we had expected. If a model performs well according to the MAP and bad according to the ABX, which metric should be trusted? For research in this field to go forward, there is a need to quantify the correlation of these two metrics.

In this paper, we wanted to go further and check that MAP and ABX can also predict performances on a downstream task. Such tasks are numerous, but one of them has not yet received enough interest: the *unsupervised frequency estimation*. We define the frequency of a speech sequence as the number of times the phonetic transcription of this sequence appears in the corpus. When dealing with text corpora, frequencies can be computed exactly with a lookup table and are used in many NLP applications. In the absence of labels, deriving the frequency of a speech sequence becomes a problem of density estimation. Estimated frequencies can be useful in representation learning by enabling efficient sampling of tokens in a speech database [7]. Also, frequencies could be used for the unsupervised word segmentation using algorithms similar to those used in text [19].

In Section 2, we present the range of embedding models that can be grouped in five categories of increasing expected reliability: hand-crafted, unsupervised, self-supervised, supervised plus a top-line embedding. In Section 3, we present the MAP and ABX metrics and introduce our frequency estimation task. In Section 4, we present results on the five speech datasets from the ZeroSpeech Challenge [2–4]. From these results, we draw guidelines for future improvements in the field of acoustic speech embeddings.

2. Embedding Methods

2.1. Acoustic features

Neural networks learn representations on top of input features. Therefore we used two types of acoustic features known as the log-MEL filterbanks (Mel-F) [24] and the Perceptual Linear Prediction (PLP) [25]. These two features can be considered as two levels of phonetic abstraction: a high-level one (PLP) and a low-level one (Mel-F). Formally, let us define a speech sequence s_t by x_1,x_2,...,x_T, where $x_i \in \mathbb{R}^n$ is called a frame of the acoustic features. T is the number frames in the sequence s_t. In our setting, these frames are spaced out every 10 ms each representing a 25 ms span of the raw signal.

2.2. Hand-crafted model: Gaussian downsampling

Holzenberger and al. [6] described a method to create fixed-size embedding vectors that requires no training of neural net-

works: the Gaussian down-sampling (GD). Given a sequence s_t, l equidistant frames are sampled and a Gaussian average is computed around each sample. It returns an embedding vector e_t of size $l \times n$ for any size T of input sequences. Therefore, given our two acoustic features, two baselines model are derived: the Gaussian-down-sampling-PLP (GD-PLP) and the Gaussian-down-sampling-Mel-F (GD-Mel-F).

Similarly, we derived a simple top-line model. Instead of using hand-crafted features, we can use the transcription of a given random segment. Each frame x_i in a sequence s_t will be assigned a 1-hot vector referring directly to the phoneme being said. This model goes through the same Gaussian averaging process to form the Gaussian-down-sampling-1hot (GD-1hot) model. This model is almost the true labels notwithstanding the information loss due to compression.

2.3. Unsupervised model: RNNAE

A more elaborate way to create speech embeddings is to learn them on top of acoustic features using neural networks. Specifically, recurrent neural networks (RNN) can be trained with back-propagation in an auto-encoding (AE) objective: the RNNAE [6, 10]. Formally, the model is composed of an encoder network, a decoder network and a speaker encoder network. The encoder maps s_t to e_t, a fixed-size vector. The speaker encoder maps the speaker identity to a fixed size vector spk_t. Then, the decoder concatenate e_t and spk_t and maps them to $\hat{s}_t$, a reconstruction of s_t. The three networks are trained jointly to minimize the *Mean Square Error* between $\hat{s}_t$ and s_t.

2.4. Self-supervised and supervised models: CAE, Siamese and CAE-Siamese

2.4.1. Advanced training objectives

We consider two popular embedding models. They are also encoder-decoders but they use additional side information. One is trained according to the Siamese objective [7,26] the other is a correspondence auto-encoder (CAE) objective [12]. Both models assume a set of pairs of sequences from the training corpus. Positive pair are assumed to have the same transcription, negative pairs, different transcriptions. Let $p_t = (s_t, s_{t'}, y)$ where $(s_t, s_{t'})$ is a pair of sequences of lengths T and T'. A binary value y indicates the positive or negative nature of the pair. We will see how to find such pairs in the next sub-section.

The CAE objective uses only positive pairs. The auto-encoder is asked to encode s_t into e_t and decode it into $\hat{s}_t$. The speaker encoder network is used similarly as for the RNNAE. To satisfy the CAE objective, $\hat{s}_t$ has to minimise the *Mean Square Error* between $\hat{s}_t$ and s'_t. It forces the auto-encoder to learn a common representation for s_t and $s_{t'}$.

The Siamese objective does not need the decoder network. It encodes both s_t and $s_{t'}$ and forces the encoder to learn a similar or different representation depending on whether the pair is positive or negative.

$$L_s(e_t, e_{t'}, y) = y \cos(e_t, e_{t'}) - (1-y) \max(0, \cos(e_t, e_{t'}) - \gamma)$$

where cos is the cosine similarity and γ is a margin. This latter accounts for negative pairs whose transcriptions have phonemes in common. These pairs should not have embeddings 'too' far away from each other. The CAE and Siamese objective can also be combined into a CAE-Siamese loss by a weighted sum of their respective loss function [27].

2.4.2. Finding and choosing pairs of speech embeddings

Finding positive pairs of speech sequences is an area of research called *unsupervised term discovery* (UTD) [16–18, 28]. Such UTD systems can be DTW alignment based [16] or involve a k-Nearest-Neighbours search [28]. We opted for the latter, as it is both scalable and among the state-of-the-art methods. It encodes exhaustively all possible speech sequences with an embedding model, and used optimized K-NN search [29] to retrieve acoustically similar pairs of speech sequences (see the details in [28]). In our experiments, we used the pairs retrieved by k-NN on *GD-PLP* encoded sequences to train our self-supervised models (CAE,Siamese, CAE-Siamese).

As a supervised alternative, it is possible to sample 'gold' pairs, i.e pairs of elements that have the exact same transcription. These 'gold' pairs are given to the CAE, Siamese and CAE-Siamese to train supervised models. These supervised models indicate how good these self-supervised models could be if we enhanced the UTD system.

3. Evaluation metrics and frequency estimation

3.1. Intrinsic quality metrics: ABX and MAP

The intrinsic quality of an acoustic speech embedding can be measured using two types of discrimination tasks: the MAP (also called same-different) [22] and ABX tasks [23]. Let us consider a set of n acoustic speech embeddings: $((e_1, t_1), (e_2, t_2), ...(e_n, t_n))$ where e_i are the embeddings and t_i the transcriptions. The ABX task creates all possible triplets (e_a,e_b,e_x) such that: $t_a = t_x$ and $t_b \neq t_x$. The model is asked to predict 1 or 0 to indicate if e_x is of type t_a or t_b. Such triplets are instances of the phonetic contrast between t_a and t_b. Formally for a given a triplet, the task is to predict:

$$y(e_x, e_a, e_b) = \mathbb{1}_{d(e_a,e_x) \leq d(e_b,e_x)}$$

The error rate on this classification task is the ABX score. It is first averaged by type of contrast (all triplets having the same t_a and t_b) then average over all contrasts.
The MAP task forms a list of all possible pairs of embeddings (e_a,e_x). The model is asked to predict 1 or 0 to indicate if e_x and e_a have the same type, i.e the same transcription, or not. Formally for a given pair, the model predicts:

$$y(e_a, e_x, \theta) = \mathbb{1}_{d(e_a,e_x) \leq \theta}$$

The precision and recall on this classification task are computed for various values of θ. The final score or the MAP task is obtained by integrating over the precision-recall curve.

3.2. Downstream task: unsupervised frequency estimation

3.2.1. The R^2 metric

Here, we introduce the novel task of frequency estimation as the assignment, for each speech sequence, of a positive real value that correlate with how frequent the transcription of this sequence is in a given reference corpus[2]. To evaluate the quality of frequency estimates, we use the correlation determinant R^2 between estimation and true frequencies. We compute this number in log space, to take into account the power-law distribution of frequencies in natural languages [30]. This coefficient

[2]This estimation could be up to a scaling coefficient; the task of finding exact count estimates is a harder task, not tackled in this paper.

is between 0 and 1 and tells what percentage of variance in the true frequencies can be explained by the estimated frequencies.

3.2.2. k-NN and density estimation

We propose to estimate frequencies using density estimation, also called the Parzen-Rosenblatt window method [31]. Let N be the number of speech sequence embeddings. First, these N embeddings are indexed into a k-NN graph, noted G, where all distances between embeddings are computed. Then, for each embedding, we search for the k closest embeddings in G. Formally, given an embedding e_t from the k-NN graph G, we compute its k distances to its k closest neighbours (d_{n_1},...d_{n_k}). The frequency estimation is a density estimation function κ of the k-NN graph G that has three parameter: a Gaussian kernel β, the number of neighbours k and the embedding e_t.

$$\kappa_G(e_t, \beta, k) = \sum_{i=1}^{k} \exp^{-\beta d_{n_i}^2}$$

This density estimation yields a real number in $[1, k]$, which we take as our frequency estimation. We set k to 2000, the maximal frequency that should be predicted using the transcription of our training corpus (the Buckeye). Then, we must tune β, the dilation of the space of a given embedding model. For each model, we choose β such that it maximises the variance of the estimated log frequencies, thereby covering the whole spectrum of possible log frequencies, in our case $[0, \log(k)]$, which is beneficial for power-law types of distribution. Note that the β kernel cannot be too large (resp. small), as it would predict only high (resp. low) values.

3.2.3. Density estimation versus clustering

Models/methods	K-means	HC-K-means	k-NN
GD-1hot	0.67	0.73	**0.74**
RNNAE Mel-F	0.30	0.35	**0.41**
CAE Siamese Mel-F	0.26	0.37	**0.43**

Table 1: *Frequency estimations using K-means, HC-K-means and k-NN density estimation on a subset of the Buckeye corpus*

We compared density estimation with an alternative method: the clustering of speech embeddings. Jansen et al. [32] did a thorough benchmark of clustering methods on the task of clustering speech embeddings. Across all their metrics, the model that performs best is Hierarchical-K-means (HC-K-means), an improved version of K-means for a higher computational cost. HC-K-means is not scalable to our data sets, so we extracted 1% of the Buckeye corpus in order to compare it with our method. A similar size of corpus is used by Jansen et al. [32].

We applied k-NN, K-means and HC-K-means from the python library scikit-learn [33] on three of our models on this subset. For K-means and HC-K-means, we used the hyper-parameters that gave the best scores in [32], namely k-means++ initialisation and average linkage function for HC-K-means. On our subset, the ground truth number of clusters is $K = 33000$. Yet, we did a grid-search on the value of k that maximises the R^2 score for frequency estimation. We found that K-means and HC-K-means perform better for $K = 20000$. It shows these algorithms are not tuned to handle data distributed according to the Zipf's law. Indeed K-means is subject the so-called 'uniform effect' and tends to find clusters of uniform sizes [34]. Table 1 shows that even by optimizing the number of clusters K, the k-NN method outperforms K-means and HC-K-means.

4. Experiments

4.1. Data sets

Five data sets at our disposal from the ZeroSpeech challenge: Conversational English (a sample of the Buckeye [35] corpus), English, French, Xitsonga and Mandarin [2,3]. These are multi-speaker non-overlapping (i.e one speaker per file) recordings of speech. All silences were removed using *voice activity detection* and corrected manually.

Each corpus was split into all possible segmentations to produce random speech sequences as described in [28]. Random speech sequences span from $70ms$ to $1s$. Shorter than $70ms$ sequences may contain less than one phoneme or be ill-pronounced phonemes. Therefore we removed very short sequences to avoid issues that are out of scope of this study.

The Buckeye sample corpus contains 12 speakers and 5 hours of speech. The French and English corpora being much larger, we reduced their number of speech sequences and speakers to the size of the Buckeye. Mandarin and Xitsonga are smaller data sets and were left untouched.

4.2. Training and hyperparameters

Our encoder-decoder network is a specific use of a three-layers bi-directional LSTM as described by Holzenberger et al. [6] with hyper-paremeters selected to minimize the ABX error on the Buckeye corpus. The speaker embedding network is a single fully connected layer with fifteen neurons. Our UTD system [28] uses the embeddings of the GD-PLP model. A set of speech pairs is returned, sorted by cosine similarity. We selected the pairs that have a cosine similarity above 0.85 as it seemed to be optimal on the Buckeye corpus according to the ABX metric. In comparison, we trained our supervised models with 'gold' pairs, i.e pairs with the exact same transcription.

Each corpus was randomly split into train (90%), dev(5%) and test (5%). Neural networks were trained on the train set, early stopping was done using the dev set and metrics computed on the test set. Specifically, we trained each model on the five training sets using the Buckeye's hyper-parameters.

MAP was ABX were computed on the test sets. Frequency estimation was computed by indexing the five training sets and building k-NN graphs. For each element of a given test set, we searched neighbours and estimated frequencies using the k-NN graphs. We used the FAISS [29] library that provides an optimised k-NN implementation.

4.3. Results

4.3.1. Across models

The results of the two metrics and downstream task are shown in Figure 1 and the following broad trends can be observed.

- Supervised models yield substantially lower performance than the ground truth 1-hot encodings, on all metric and all languages. These supervised models have a margin for improvement as they do not learn optimal embeddings despite having access to ground truth labels.
- Supervised models outperform their corresponding self-supervised model, in almost all metrics and for all languages. It means that self-supervision has also a margin for improvement given better pairs from the UTD systems.

		R^2 on frequency estimation (100 is best score)						MAP (100 is best score)						ABX (0 is best score)					
		Buckeye	English	Xitsonga	French	Mandarin	*average*	Buckeye	English	Xitsonga	French	Mandarin	*average*	Buckeye	English	Xitsonga	French	Mandarin	*average*
topline	GD 1hot	87	86,2	86	80	86,4	*85,1*	82,5	84,1	98,9	89,7	89,3	*88,9*	0,1	0,1	0,1	0,1	0	0,1
supervised	CAE Siamese Mel-F gold	57,9	59,1	61,5	64	53,5	*59,2*	25,9	25,2	87,8	46,6	27,8	*42,7*	1,6	1,5	1,3	2,1	1,5	1,6
	CAE Siamese PLP gold	59,6	60,3	62,2	61	52	*59*	32,2	28,6	90,4	38,2	28,4	*43,6*	1,3	1,2	1,3	2,2	1,3	1,5
	Siamese Mel-F gold	42,3	59,3	58	59	49,9	*53,7*	16,9	25,7	88,5	37,3	20,8	*37,8*	2,3	1,7	1,7	2,6	1,8	2
	Siamese PLP gold	33,1	58,5	51,5	57	49,9	*50*	15,4	26,3	85,6	34,2	25,2	*37,3*	2,6	1,6	1,8	3,2	1,4	2,1
	CAE Mel-F gold	59,4	60	69,9	53	48,3	*58,1*	20,2	7,3	61,1	4,8	17,7	*22,2*	3,1	3,2	3,5	2,8	4	3,3
	CAE PLP gold	46,4	66,8	72,5	54	65,4	*61*	29,7	8,6	54,9	5,5	22,5	*24,2*	1,5	2,4	3,3	2,6	1,7	2,3
self-supervised	CAE Siamese Mel-F	60,2	58	45,3	60	59,5	*56,6*	18,3	9,6	50,1	16,1	19,1	*22,6*	5,3	6,6	7,8	4,7	5,2	5,9
	CAE Siamese PLP	45	47	37,9	50	59,5	*47,9*	16,6	11,1	46,2	11,7	19,6	*21*	7	6,5	8,8	5,4	5,9	6,7
	Siamese Mel-F	25,5	47,2	46,8	50	54,1	*44,7*	11,2	9,8	52,1	11,7	15,8	*20,1*	6,2	7,9	8	9,8	6,2	7,6
	Siamese PLP	38,8	17,5	24,4	39	39,2	*31,8*	16	7	42,7	7	17	*17,9*	8	7,9	9,2	13	6,2	8,9
	CAE Mel-F	48,9	0	17	10	27,4	*20,7*	11	4,1	21,8	2,9	15,5	*11,1*	4,8	7,7	14,1	5,9	7,8	8,1
	CAE PLP	1,2	3,1	49,4	0	15,4	*13,8*	7,1	4,9	24,9	3	14,9	*11*	5,4	6,3	13,6	7,1	5,5	7,6
un-supervised	RNNAE Mel-F	51,5	58,3	70,8	57	59,5	*59,4*	8,6	4,7	36,5	3,9	14,1	*13,6*	6	3,1	6,6	2,9	4,5	4,6
	RNNAE PLP	0,8	39,7	61,1	1	59,4	*32,4*	9,3	7	42,5	5,4	19,2	*16,7*	9,2	3,8	7,5	4,1	2,3	5,4
hand-crafted	GD Mel-F	15,5	24,1	19,5	41	32,6	*26,5*	11,4	9,4	38,4	17,8	18,7	*19,1*	11,7	8,9	14,2	15,5	8,3	11,7
	GD PLP	41,7	42,8	40,6	49	41,7	*43,2*	17,6	11,2	45,7	14,7	24,9	*22,8*	9,8	7,9	12,5	13,4	7,8	10,3

Figure 1: *Value of metrics and the downstream task across models, corpora. The average column is the average score over all corpora*

- Among self-supervised and supervised models, the CAE-Siamese Mel-F takes the pole position. This model seems to be able to combine the advantages of both training objectives. A result already claimed by [27].
- (self) supervised neural neural networks trained on low-level acoustic features (Mel-F) performs better or equally well as high-level acoustic features (PLP). This shows that neural networks can learn their own high-level acoustic features from low-level information.
- Self-supervised models are expected to outperform unsupervised models because they use side information. Yet many configurations do not show this consistently. Only the Buckeye data set seems consistent, but this dataset is the one on which pairs were selected through a grid-search to minimise the ABX error. This may be due to the variable quality of the pairs found by UTD; better UTD is therefore needed to help self-supervised models.
- Unsupervised models are supposed to be better than hand-crafted models because they can adjust by learning from the dataset. Yet, this is not consistently found. Hand crafted models are worse than unsupervised models for ABX and frequency estimation but not for MAP.
- In detail, which model is best in a particular language depends on the metric.

4.3.2. Across metrics and frequency estimation

subsubsectionInterpretation of results across metrics and frequency estimation

We quantified the possibility to observe the discrepancies that we have just discussed in Table 2. We computed the correlation R^2 across the three 'average' columns. Cross correlation scores range from $R^2 = 0.33$ to 0.53; the top-line model is not included when computing these scores.

R^2	Frequency est.	MAP	ABX
Frequency est.	1.0	0.34	0.53
MAP	0.34	1.0	0.45
ABX	0.53	0.45	1.0

Table 2: *Correlation R^2 across the 'average' column of MAP, ABX and frequency estimation*

These correlations are low enough to permit sizeable discrepancies across metrics and the downstream task. One of our model, the RNNAE Mel-F, epitomises the problem. This model is comparatively bad according to the MAP but good according to ABX and the frequency estimation. It means that MAP and ABX reveal different aspect of the reliability of embedding models. Therefore, only large progress according to one metric assures a progress according to an other metric. It shows the limit of intrinsic evaluation of speech embeddings. Moderate variations on a intrinsic metric cannot guarantee a progress on a given downstream task.

ABX and MAP scores are averages over multiple phonetic contrasts. These contrasts could be clustered based on their phonetic frequencies, average lengths or number of phonemes in common. Such fined-grained analyses can sometimes understanding divergences across metrics. However, we have been unable to find a categorisation of results that make sense of Figure 1 as a whole. There are currently no fully reliable metrics to assess the intrinsic quality of speech embeddings.

5. Conclusion

We quantified the correlation across two intrinsic metrics (MAP and ABX) and a novel downstream task: frequency estimation. Although MAP and ABX agree on general categories (like supervised versus unsupervised embeddings), we also found large discrepancies when it comes to select a particular model highlighting the limits of these intrinsic quality metrics. However convenient intrinsic metrics may be, they only show partial views of the overall reliability of a model. We showed using frequency estimation that variations on intrinsic quality metrics should not be accounted for certain progress on downstream tasks. More attention should be brought on downstream tasks that have the credit to answer practical problems.

6. Acknowledgements

We thanks Matthijs Douze for useful comments on density estimation. This work was funded in part by the Agence Nationale pour la Recherche (ANR-17-EURE-0017 Frontcog, ANR-10-IDEX-0001-02 PSL*, ANR-19-P3IA-0001 PRAIRIE 3IA Institute), CIFAR, and a research gift by Facebook.

7. References

[1] A. Jansen, E. Dupoux, S. Goldwater, M. Johnson, S. Khudanpur, K. Church, N. Feldman, H. Hermansky, F. Metze, R. Rose, M. Seltzer, P. Clark, I. McGraw, B. Varadarajan, E. Bennett, B. Borschinger, J. Chiu, E. Dunbar, A. Fourtassi, D. Harwath, C. Lee, K. Levin, A. Norouzian, V. Peddinti, R. Richardson, T. Schatz, and S. Thomas, "A summary of the 2012 jhu clsp workshop on zero resource speech technologies and models of early language acquisition," in *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, 2013, pp. 8111–8115.

[2] M. Versteegh, R. Thiollière, T. Schatz, X.-N. Cao Kam, X. Anguera, A. Jansen, and E. Dupoux, "The zero resource speech challenge 2015," 09 2015.

[3] E. Dunbar, X. Cao, J. Benjumea, J. Karadayi, M. Bernard, L. Besacier, X. Anguera, and E. Dupoux, "The zero resource speech challenge 2017," *CoRR*, vol. abs/1712.04313, 2017. [Online]. Available: http://arxiv.org/abs/1712.04313

[4] E. Dunbar, R. Algayres, J. Karadayi, M. Bernard, J. Benjumea, X.-N. Cao, L. Miskic, C. Dugrain, L. Ondel, A. Black, L. Besacier, S. Sakti, and E. Dupoux, "The zero resource speech challenge 2019: Tts without t," 04 2019.

[5] H. Kamper, "Truly unsupervised acoustic word embeddings using weak top-down constraints in encoder-decoder models," *CoRR*, vol. abs/1811.00403, 2018. [Online]. Available: http://arxiv.org/abs/1811.00403

[6] N. Holzenberger, M. Du, J. Karadayi, R. Riad, and E. Dupoux, "Learning word embeddings: Unsupervised methods for fixed-size representations of variable-length speech segments," 09 2018, pp. 2683–2687.

[7] R. Riad, C. Dancette, J. Karadayi, N. Zeghidour, T. Schatz, and E. Dupoux, "Sampling strategies in siamese networks for unsupervised speech representation learning," *CoRR*, vol. abs/1804.11297, 2018. [Online]. Available: http://arxiv.org/abs/1804.11297

[8] A. L. Maas, S. D. Miller, T. M. O'neil, and A. Y. Ng, "Word-level acoustic modeling with convolutional vector regression."

[9] K. Levin, K. Henry, A. Jansen, and K. Livescu, "Fixed-dimensional acoustic embeddings of variable-length segments in low-resource settings," 12 2013, pp. 410–415.

[10] Y.-A. Chung, C.-C. Wu, C.-H. Shen, H.-y. Lee, and L.-S. Lee, "Audio word2vec: Unsupervised learning of audio segment representations using sequence-to-sequence autoencoder," 03 2016.

[11] Y. Chung, W. Weng, S. Tong, and J. R. Glass, "Unsupervised cross-modal alignment of speech and text embedding spaces," *CoRR*, vol. abs/1805.07467, 2018. [Online]. Available: http://arxiv.org/abs/1805.07467

[12] 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," *Annual Conference of the International Speech Communication Association*, 2015.

[13] T. J. Hazen, W. Shen, and C. White, "Query-by-example spoken term detection using phonetic posteriorgram templates," in *2009 IEEE Workshop on Automatic Speech Recognition Understanding*, 2009, pp. 421–426.

[14] K. Levin, A. Jansen, and B. Van Durme, "Segmental acoustic indexing for zero resource keyword search," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2015, pp. 5828–5832.

[15] Y. Wang, H. Lee, and L. Lee, "Segmental audio word2vec: Representing utterances as sequences of vectors with applications in spoken term detection," *CoRR*, vol. abs/1808.02228, 2018. [Online]. Available: http://arxiv.org/abs/1808.02228

[16] A. S. Park and J. R. Glass, "Unsupervised pattern discovery in speech," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 16, no. 1, pp. 186–197, 2008.

[17] C.-y. Lee, T. J. O'Donnell, and J. Glass, "Unsupervised lexicon discovery from acoustic input," *Transactions of the Association for Computational Linguistics*, vol. 3, pp. 389–403, 2015. [Online]. Available: https://www.aclweb.org/anthology/Q15-1028

[18] O. Räsänen, G. Doyle, and M. Frank, "Unsupervised word discovery from speech using automatic segmentation into syllable-like units," 09 2015.

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

[20] H. Kamper, A. Jansen, and S. Goldwater, "A segmental framework for fully-unsupervised large-vocabulary speech recognition," *Computer Speech and Language*, 2017.

[21] K. Kawakami, C. Dyer, and P. Blunsom, "Unsupervised word discovery with segmental neural language models," *CoRR*, vol. abs/1811.09353, 2018. [Online]. Available: http://arxiv.org/abs/1811.09353

[22] M. A. Carlin, S. Thomas, A. Jansen, and H. Hermansky, "Rapid evaluation of speech representations for spoken term discovery," in *INTERSPEECH*, 2011.

[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," *INTERSPEECH 2013: 14th Annual Conference of the International Speech Communication Association*, pp. 1–5, 01 2013.

[24] S. B. Davis and P. Mermelstein, "Comparison of parametric representations for monosyllabic word recognition in continuously spoken sentences," *ACOUSTICS, SPEECH AND SIGNAL PROCESSING, IEEE TRANSACTIONS ON*, pp. 357–366, 1980.

[25] H. Hermansky, "Perceptual linear predictive (plp) analysis of speech." *The Journal of the Acoustical Society of America*, vol. 87 4, pp. 1738–52, 1990.

[26] R. Thiollière, E. Dunbar, G. Synnaeve, M. Versteegh, and E. Dupoux, "A hybrid dynamic time warping-deep neural network architecture for unsupervised acoustic modeling," in *INTERSPEECH*, 2015.

[27] P. Last, H. A. Engelbrecht, and H. Kamper, "Unsupervised feature learning for speech using correspondence and siamese networks," *IEEE Signal Processing Letters*, vol. 27, pp. 421–425, 2020.

[28] A. Thual, C. Dancette, J. Karadayi, J. Benjumea, and E. Dupoux, "A k-nearest neighbours approach to unsupervised spoken term discovery," in *2018 IEEE Spoken Language Technology Workshop (SLT)*, 2018, pp. 491–497.

[29] J. Johnson, M. Douze, and H. Jégou, "Billion-scale similarity search with gpus," *arXiv preprint arXiv:1702.08734*, 2017.

[30] G. K. Zipf, *The psycho-biology of language*, 1935.

[31] E. Parzen, "On estimation of a probability density function and mode," *Ann. Math. Statist.*, vol. 33, no. 3, pp. 1065–1076, 09 1962. [Online]. Available: https://doi.org/10.1214/aoms/1177704472

[32] H. Kamper, A. Jansen, S. King, and S. Goldwater, "Unsupervised lexical clustering of speech segments using fixed-dimensional acoustic embeddings," in *2014 IEEE Spoken Language Technology Workshop (SLT)*, 2014, pp. 100–105.

[33] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[34] J. Wu, *The Uniform Effect of K-means Clustering*, 07 2012, pp. 17–35.

[35] M. A. Pitt, K. Johnson, E. Hume, S. Kiesling, and W. Raymond, "The buckeye corpus of conversational speech: Labeling conventions and a test of transcriber reliability," *Speech Communication*, vol. 45, no. 1, pp. 89–95, 2005.