Improved acoustic word embeddings for zero-resource languages using multilingual transfer

Herman Kamper Yevgen Matusevych Sharon Goldwater

***Abstract*—Acoustic word embeddings are fixed-dimensional representations of variable-length speech segments. Such embeddings can form the basis for speech search, indexing and discovery systems when conventional speech recognition is not possible. In *zero-resource* settings where unlabelled speech is the only available resource, we need a method that gives robust embeddings on an arbitrary language. Here we explore multilingual transfer: we train a single supervised embedding model on labelled data from multiple well-resourced languages and then apply it to unseen zero-resource languages. We consider three multilingual recurrent neural network (RNN) models: a classifier trained on the joint vocabularies of all training languages; a Siamese RNN trained to discriminate between same and different words from multiple languages; and a correspondence autoencoder (CAE) RNN trained to reconstruct word pairs. In a word discrimination task on six target languages, all of these models outperform state-of-the-art unsupervised models trained on the zero-resource languages themselves, giving relative improvements of more than 30% in average precision. When using only a few training languages, the multilingual CAE performs better, but with more training languages the other multilingual models perform similarly. Using more training languages is generally beneficial, but improvements are marginal on some languages. We present probing experiments which show that the CAE encodes more phonetic, word duration, language identity and speaker information than the other multilingual models.**

***Index Terms*—acoustic word embeddings, multilingual models, transfer learning, zero-resource speech processing.**

I. Introduction

The dependence of automatic speech recognition (ASR) systems on transcribed speech data remains a major hurdle for developing speech technology in new languages. Researchers in *zero-resource speech processing* aim to develop unsupervised methods that can learn directly from unlabelled speech audio [1]–[3]. Several tasks have been tackled. In unsupervised term discovery (UTD), the goal is to find recurring word- or phrase-like patterns in an unlabelled speech collection [4]. In query-by-example search, the aim is to identify utterances containing instances of a given spoken query [5]–[7]. Full-coverage segmentation and clustering aims to tokenise an entire speech set into word-like units [8]–[11].

In these applications, speech segments of different durations need to be compared. One approach is to use dynamic time warping (DTW), but this is computationally expensive and can be inaccurate [12]. Levin et al. [13] therefore proposed an alignment-free approach: a speech segment of arbitrary length is embedded in a fixed-dimensional space. The resulting vectors are referred to as *acoustic word embeddings*. Segments can then be compared by simply calculating a distance between their vectors in the embedding space. This requires a mapping such that instances of the same word type (lexical item) have similar embeddings. Several supervised and unsupervised acoustic word embedding methods have since been proposed [14]–[18].

While unsupervised methods are useful in that they can be used in zero-resource settings, there is still a large performance gap compared to supervised methods [13], [18]. Here we investigate whether supervised modelling can still be used to obtain accurate embeddings on a language for which no labelled data is available. Specifically, we propose to exploit labelled resources from languages where these are available, allowing us to take advantage of state-of-the-art supervised modelling methods, but to then apply the resulting model to zero-resource languages for which no labelled data is available.

For this transfer learning approach, we consider three multilingual acoustic embedding models. A classifier recurrent neural network (RNN) is trained on the joint vocabularies of several well-resourced languages [19]. A Siamese RNN is trained with a contrastive loss on multiple languages so that word instances of the same type have similar embeddings while others are pushed away [19]. Finally, the correspondence autoencoder RNN (CAE-RNN) uses an encoder-decoder structure to reconstruct one word given another word of the same type as input [20]. We use six well-resourced languages from the GlobalPhone corpus [21] for training, and evaluate the models using a word discrimination task on six target languages which we treat as zero-resource.

Our goal is to find the best acoustic embedding approach for zero-resource languages. To show that multilingual transfer is the best in this setting, we need to compare it to an appropriate unsupervised monolingual baseline. We extend [18], where an unsupervised CAE-RNN was trained on discovered terms from a UTD system, and use the same approach to train unsupervised ClassifierRNN and SiameseRNN models on unlabelled data from zero-resource languages, similar to [22]. For the first time, these three unsupervised models are compared. The CAE-RNN performs best. To answer our main research question, we compare this unsupervised approach to the supervised multilingual CAE-RNN, ClassifierRNN and SiameseRNN. This is an extension of [23], where the first two were considered. The SiameseRNN—a popular method in the monolingual supervised setting [19]—has not been previously used for

H. Kamper is with the Department of E&E Engineering, Stellenbosch University, South Africa (email: kamperh@sun.ac.za).

Y. Matusevych and S. Goldwater are with the School of Informatics, University of Edinburgh, UK (email: ymatusev@ed.ac.uk and sgwater@inf.ed.ac.uk).

We thank C. Jacobs for helpful input. This work is based on research supported in part by the National Research Foundation of South Africa (grant number: 120409), a James S. McDonnell Foundation Scholar Award (220020374), an ESRC-SBE award (ES/R006660/1), and a Google Faculty Award for HK.

multilingual transfer. Across the six zero-resource languages, all three multilingual models outperform unsupervised CAE-RNNs trained on the respective evaluation languages.

Of the three approaches, the multilingual CAE-RNN performs better when only a few well-resourced training languages are used, but with more languages there is little difference between the models. Here we also introduce a new variant of the multilingual CAE-RNN which conditions its decoder on the training language identity. This gives small but consistent improvements on the zero-resource languages. Finally, inspired by the analysis of supervised monolingual acoustic word embedding models presented in [24], we perform probing experiments to see how the multilingual models organise their embedding spaces. We find that, compared to the other multilingual models, the CAE-RNN captures more phonetic, word duration and speaker information. Source code is released at `https://github.com/kamperh/globalphone_awe`.

II. Related Work

Acoustic word embeddings are fixed-dimensional vector representations of arbitrary-length spoken words [13]. The goal is to map different instances of the same spoken word to similar embeddings, while mapping words of different types to embeddings that are far apart. This allows variable-length speech segments to be efficiently compared directly in a fixed-dimensional embedding space without any alignment (which can be slow). Acoustic embeddings are therefore being used increasingly in downstream tasks such as full-coverage segmentation and clustering [11], query-by-example speech search [7], [25], [26], and spoken content retrieval [17], [27].

Our goal is to learn an acoustic word embedding model that can be applied in a zero-resource setting where no labelled data is available in the target language. Unsupervised modelling can be used directly in this setting. A simple but effective early unsupervised approach is *downsampling* [13], [28]: a fixed number of equally spaced frames are used to represent a speech segment. More recent work has turned to neural methods. Unsupervised auto-encoding recurrent neural networks (RNNs) use an encoder-decoder structure to try to reconstruct variable-duration input [14], [17], [28]. Instead of trying to reconstruct the input exactly, the correspondence autoencoder RNN (CAE-RNN) [18] is trained to reconstruct another speech segment predicted to be of the same type as the input. Since labelled data is not available, a UTD system—itself unsupervised—is used to find word-like pairs predicted to be of the same unknown type. This model outperformed downsampling and an AE-RNN in [18]. However, it still falls far short from supervised models trained on labelled data. Here we investigate whether supervised modelling can still be used to obtain embeddings on a language for which no labelled data is available.

Early supervised acoustic word embedding methods relied on a reference vector approach: the DTW distances between an input segment and a fixed reference set are calculated, followed by supervised dimensionality reduction [13]. This has since been outperformed by supervised neural approaches relying on convolutional [15], [29] and RNN-based [16], [19] architectures. These networks can be trained with a classification loss on labelled isolated words, where features from an intermediate layer before the final softmax are used as embeddings, or with a contrastive loss, where a Siamese network with tied branches explicitly pushes embeddings of words of the same type together while pushing different words apart. Most studies have reported that the contrastive loss performs better, although improvements over the classifier are sometimes small [15], [19]. More recent work has started to incorporate surrounding context [26], [30], [31], in some cases specifically to capture semantic relationships [27], [32]. Some models explicitly embed written and spoken words together [17], [33]–[37]. Although these directions are important, we focus on the original question of learning embedding models that map together spoken words of the same type.

In all of the above studies, models were always subsequently applied to the *same language* as the one used during training. One approach we consider here is to train a supervised model on (multiple) well-resourced languages and then apply it to a zero-resource language. Our aim is to determine the best embedding approach when confronted with a zero-resource setting: is it better to apply a (potentially multilingual) supervised model on an unseen language, or is it better to train in an unsupervised fashion on the target language itself? Apart from studies explicitly looking at acoustic embeddings, this question also relates to several other lines of research.

Most importantly, our work is inspired by studies showing the benefit of using multilingual bottleneck features as frame-level representations for zero-resource languages [38]–[41]: a frame-level network is trained jointly on several well-resourced languages (normally to predict context-dependent triphone HMM states) and is then applied to an unseen language. In [42], multilingual data was also used for discovering acoustic units. As in these studies, our findings show the advantage of learning from labelled data in well-resourced languages when processing an unseen low-resource language—here at the word rather than subword level. We are also inspired by studies in ASR aiming to build a single system that can transcribe speech from a number of different input languages. Early approaches considered joint training of a single set of HMM-GMM models on multiple languages [43], [44], while more recent work has turned to end-to-end neural networks [45]–[48]. While these models are also typically trained jointly on multiple languages (as we do here), they are then applied to test data from languages on which they are trained. In contrast, the languages to which our models are applied are never seen during training.

Training a machine learning model on one task and then applying it to another related problem is becoming an attractive way to leverage existing data or models—a methodology referred to as *transfer learning* [49]. Our approach can be described as multilingual transfer since we train a single embedding model on labelled data from several well-resourced languages and then embed speech from a target zero-resource language. While in many transfer learning settings there is a subsequent fine-tuning step with some (limited) labelled data from the target domain (referred to as *inductive transfer learning*), we directly apply a multilingual model to a new language (called *transductive transfer learning* in [49], [50]).

III. Acoustic Word Embedding Models

For obtaining acoustic word embeddings on a zero-resource language, we compare unsupervised models trained on within-language unlabelled data to supervised models trained on pooled labelled data from multiple well-resourced languages. We consider three different recurrent neural network (RNN) architectures. All three uses gated recurrent units [51]. Below we first describe how each architecture is used for supervised multilingual modelling, and then explain how the same architectures are used for unsupervised monolingual modelling.

A. Supervised multilingual acoustic word embeddings

Given labelled data from several well-resourced languages, we consider three supervised multilingual acoustic embedding models. We use $X = \mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_T$ to denote a sequence of frame-level acoustic feature vectors, with each vector $\mathbf{x}_t \in \mathbb{R}^D$, e.g. D-dimensional MFCCs.

1) Classifier RNN: Given a true isolated word segment X from any of the training languages, the ClassifierRNN predicts the word type of that segment. Formally, it is trained using the multiclass log loss, given as $J(X) = -\sum_{k=1}^{K} y_k \log f_k(X)$ for a single training example, where K is the size of the joint vocabulary over all the training languages, $y_k \in \{0, 1\}$ is an indicator for whether X is an instance of word type k, and $\boldsymbol{f}(X) \in [0, 1]^K$ is the predicted distribution over the joint vocabularies of all the languages. As shown in Figure 1, an input sequence X is presented to an encoder RNN shared between all training languages. A fixed-dimensional acoustic word embedding $\mathbf{z} \in \mathbb{R}^M$ is then obtained by transforming the final hidden state of the RNN [17]. This embedding is fed into a softmax layer to produce $\boldsymbol{f}(X)$. Embeddings can therefore be obtained for speech segments from a language not seen during training.

2) Siamese RNN: In the ClassifierRNN we obtain embeddings from an intermediate layer with the hope that instances of the same word type will have similar embeddings. Rather than doing this implicitly through a classification loss, the SiameseRNN explicitly optimises a similarity loss between embeddings [19], [52]. Formally, input sequences X_a, X_p, X_n are each passed through an RNN to produce embeddings $\mathbf{z}_a$, $\mathbf{z}_p$, $\mathbf{z}_n$, as illustrated in Figure 2. X_a and X_p are instances of the same word type (subscripts indicate *anchor* word and *positive* word) while X_n is a different word (*negative*). The RNN's parameters are optimised using the contrastive loss [53], [54]: $J(X_a, X_p, X_n) = \max\{0, m + d(\mathbf{z}_a, \mathbf{z}_p) - d(\mathbf{z}_a, \mathbf{z}_n)\}$, with $d(\cdot)$ the squared Euclidean distance and m a margin parameter. This loss is at a minimum when all embedding pairs $(\mathbf{z}_a, \mathbf{z}_p)$ of the same word type are more similar by a margin m than pairs $(\mathbf{z}_a, \mathbf{z}_n)$ of different types. To sample negative items we use the online semi-hard mining scheme [55]: for each positive pair in a mini-batch the most difficult negative item is used satisfying the constraint $d(\mathbf{z}_a, \mathbf{z}_d) < d(\mathbf{z}_a, \mathbf{z}_n)$, except if there is no such item in which case the negative example with the largest distance is used. The multilingual SiameseRNN is trained jointly on all the well-resourced training languages.

3) Correspondence autoencoder RNN: The CAE-RNN was originally developed as an unsupervised monolingual embedding method [18], as explained below. But given true word segments from forced alignments, it can also be trained in a supervised way. Here we train a single CAE-RNN by pooling word segments from several well-resourced training languages. Formally, the CAE-RNN is trained on pairs of speech segments (X, X'), with $X = \mathbf{x}_1, \ldots, \mathbf{x}_T$ and $X' = \mathbf{x}'_1, \ldots, \mathbf{x}'_{T'}$ containing different instances of the same word type. As illustrated in Figure 3, X is fed into an encoder RNN, which produces the acoustic word embedding $\mathbf{z}$. This embedding is used to condition a decoder RNN which attempts to

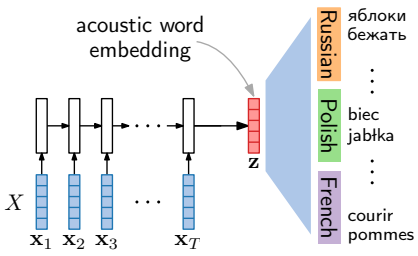

Fig. 1. The multilingual ClassifierRNN is trained jointly on all the training languages to classify which word type an input segment X belongs to. Our model is trained on data from six languages (three shown here as illustration).

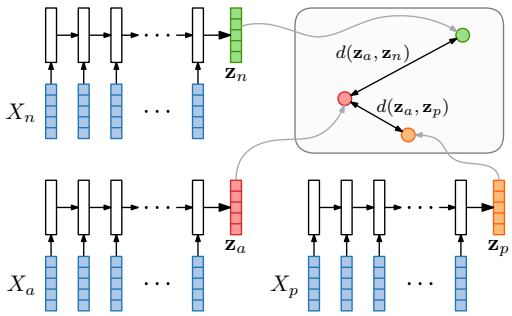

Fig. 2. The multilingual SiameseRNN is trained jointly on all the training languages so that the distance between the embedding of an anchor word $\mathbf{z}_a$ and a positive example $\mathbf{z}_p$ is smaller (by some margin) than the distance between the anchor and a negative example $\mathbf{z}_n$. The three RNNs shown in the figure share the same set of parameters.

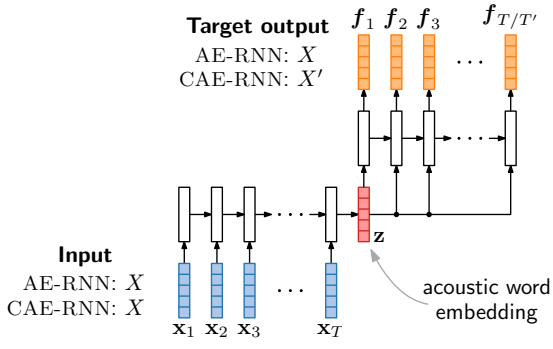

Fig. 3. The AE-RNN is trained to reconstruct its input X (a speech segment) from the latent acoustic word embedding $\mathbf{z}$. The CAE-RNN is trained to reconstruct one segment X' when presented with another segment X as input. For the supervised multilingual CAE-RNN (§III-A3), true word pairs (X, X') are used from the well-resourced training languages. For the unsupervised monolingual CAE-RNN (UTD) (§III-C1), an unsupervised term discovery system is used to find word pairs (X, X').

reconstruct X'. The loss for a single training pair is therefore $J(X, X') = \sum_{t=1}^{T'} ||\mathbf{x}'_t - \boldsymbol{f}_t(X)||^2$, where $\boldsymbol{f}_t(X)$ is the t^{th} decoder output conditioned on the embedding $\mathbf{z}$.

The idea is that CAE-RNN embeddings should be invariant to properties not common to the two segments (e.g. speaker, channel), while capturing aspects that are (e.g. word identity). Here we hope that this type of invariance can be learned from well-resourced languages and then transferred to an unseen language. This motivation for the CAE-RNN is similar to that of the SIAMESERNN. But the former relies on a softer reconstruction loss while the latter uses an explicitly contrastive loss. Also, the CAE-RNN is trained only on positive pairs, while the SIAMESERNN incorporates negative items.

B. Language conditioning

As a contribution not considered in prior work, we present model variants which use training language identity information in the decoder of a model. The idea is that this would allow a model to capture more language-specific properties after the embedding layer, while common aspects across languages are captured in the shared encoder. This could produce embeddings that generalise better to unseen languages.

Instead of having only a single shared softmax layer after the embedding layer in the multilingual CLASSIFIERRNN (§III-A1), we consider a model where we add separate fully connected layers for each training language, each language-specific branch terminating in its own softmax output layer. For the multilingual CAE-RNN (§III-A3), together with the acoustic embedding $\mathbf{z}$, we additionally append a language embedding at each decoder time-step. The language embedding matrix is updated jointly with the rest of the network.

C. Unsupervised monolingual acoustic word embeddings

An alternative to the transfer learning approaches in §III-A for obtaining acoustic embeddings on a zero-resource language is to train an unsupervised embedding model directly on unlabelled data in the target language. Four unsupervised monolingual embedding models are described below. The unsupervised CLASSIFIERRNN and SIAMESERNN have not been considered before. The experiment in §V-A is therefore also the first comparison of these unsupervised approaches.

1) Unsupervised autoencoder and correspondence autoencoder RNNs: The unsupervised autoencoding RNN (AE-RNN) of [14] is trained on unlabelled speech segments to reproduce its input. Formally, given an input speech segment X, the loss for a single training example is $J(X) = \sum_{t=1}^{T} ||\mathbf{x}_t - \boldsymbol{f}_t(X)||^2$, with $\boldsymbol{f}_t(X)$ the t^{th} decoder output, as also illustrated in Figure 3.

We next consider the unsupervised variant of the CAE-RNN, first proposed in [18]. In contrast to the supervised multilingual CAE-RNN (§III-A3), in the unsupervised setting we do not have access to transcriptions from which to construct input-output training pairs. We therefore apply an unsupervised term discovery (UTD) system [4], [56] to an unlabelled speech collection in the target zero-resource language, discovering pairs of word segments predicted to be of the same unknown type. These are used as input-output pairs (X, X') to the CAE-RNN, as shown in Figure 3. We denote the resulting unsupervised monolingual model as CAE-RNN (UTD). In all cases, we first pretrain a CAE-RNN using the AE loss above before switching to the CAE loss.

2) Unsupervised classifier and Siamese RNNs: This strategy of using a UTD system to discover training targets for the CAE-RNN (UTD) can also be employed with the other architectures of §III-A. Instead of predicting the word type of an input segment (§III-A1), the monolingual unsupervised CLASSIFIERRNN (UTD) is trained to predict the cluster labels from the UTD system. The unsupervised SIAMESERNN (UTD) uses matching pairs from the UTD system as positive examples, while sampling negative items randomly from UTD terms with different cluster labels. These two unsupervised models have not been considered in any previous work.

IV. EXPERIMENTAL SETUP

A. Data

We perform all our experiments on the GlobalPhone corpus of read speech [21]. We treat six languages as our target zero-resource languages: Spanish (ES), Hausa (HA), Croatian (HR), Swedish (SV), Turkish (TR) and Mandarin (ZH). Each language has on average 16 hours of training, 2 hours of development and 2 hours of test data. For training supervised multilingual embedding models, six other GlobalPhone languages are chosen as well-resourced languages: Czech (CS), French (FR), Polish (PL), Portuguese (PT), Russian (RU) and Thai (TH).[1] Each well-resourced language has on average 21 hours of labelled training data.

B. Model training, architectures and implementation

We pool the data from the well-resourced languages and train three supervised multilingual models (§III-A). The supervised multilingual CLASSIFIERRNN is trained jointly on true word segments from the six training languages, obtained from forced alignments. The number of word types per language is limited to 10k, giving a total of 70k output classes (more classes did not give improvements). Rather than considering all possible word pairs from all languages when training the multilingual CAE-RNN and SIAMESERNN models, we sample 300k true word pairs from the combined data. Using more pairs did not improve development performance, but increased training time.

Since we do not use transcriptions for the unsupervised monolingual embedding models (§III-C), we apply the UTD system of [56] to each of the training sets of the six zero-resource languages.[2] Roughly 36k predicted word pairs are extracted in each language. The unsupervised monolingual AE-RNN, CAE-RNN, CLASSIFIERRNN and SIAMESERNN models are all trained on the same set of UTD terms. For the AE-RNN, an alternative would have been to use segments randomly sampled from the speech audio, but UTD-discovered segments gave slightly better performance in [18].

Since development data would not be available in a zero-resource setting, we perform development experiments on

[1] In [23], Bulgarian was also used, but we decided to exclude it here because of poor-quality forced alignments (determined through manual inspection).

[2] As described at `https://github.com/eginhard/cae-utd-utils`.

labelled data from yet another language: German (DE). We used this data to tune the vocabulary size for the CLASSIFIERRNN, the number of pairs for the CAE-RNN and SIAMESERNNmodels, the margin for the SIAMESERNN, the language embedding size of the multilingual CAE-RNN, and the number of training epochs. Other hyperparameters are set as in [18].

All models are implemented in TensorFlow. Speech audio is parametrised as $D = 13$ dimensional static Mel-frequency cepstral coefficients (MFCCs). We use an embedding dimensionality of $M = 130$ throughout, since downstream systems such as the segmentation and clustering system of [11] are constrained to embedding sizes of this order. All encoder-decoder models have three encoder and three decoder unidirectional RNN layers, each with 400 units. The same encoder structure is used for the CLASSIFIERRNN and SIAMESERNN. The SIAMESERNN uses a margin of $m = 0.25$. Pairs are presented to the CAE-RNN models in both input-output directions. For the multilingual CAE-RNN using language conditioning (§III-B), a language embedding with 200 dimensions is used. Models are trained using Adam optimisation [57] with a learning rate of 0.001.

C. Evaluation and baselines

We want to measure the intrinsic quality of the resulting acoustic word embeddings without being tied to a particular downstream system architecture. We therefore use a word discrimination task designed for this purpose [58]. In the *same-different* task, we are given a pair of acoustic segments, each a true word, and we must decide whether the segments are examples of the same or different words. To evaluate a particular embedding method, a set of isolated test words are first embedded. For every word pair in this set, the cosine distance between their embeddings is calculated. Two words can then be classified as being of the same or different type based on some distance threshold, and a precision-recall curve is obtained by varying the threshold. The area under this curve is used as final evaluation metric, referred to as the average precision (AP). We are particularly interested in obtaining embeddings that are speaker invariant. As in [59], we therefore calculate AP by only taking the recall over instances of the same word spoken by different speakers (i.e., we consider the more difficult setting since a model does not get credit for recalling the same word if it is said by the same speaker).

As an additional unsupervised baseline embedding method, we use downsampling [13], [28] by keeping 10 equally-spaced MFCC vectors from a segment with appropriate interpolation, giving a 130-dimensional embedding. Finally, we report same-different performance when using DTW alignment cost to predict whether word segments are the same or not. For this baseline, we additionally include delta and double-delta MFCCs (which was found to be beneficial).

V. EXPERIMENTAL RESULTS

Our main goal is to find the best acoustic word embedding approach for an unseen zero-resource language. We hypothesise that multilingual acoustic word embedding models trained on labelled data from well-resourced languages will be superior to monolingual unsupervised embedding models trained directly on unlabelled data from the target zero-resource language. To test this hypothesis, we first need to establish the best unsupervised embedding approach to serve as an appropriate baseline. We start, therefore, in §V-A by comparing different monolingual unsupervised models. In section §V-B, we briefly consider development experiments to find the best multilingual model variants, specifically looking at the effect of language conditioning (§III-B). We then answer our main research question in §V-C by comparing the best unsupervised approach to the best multilingual embedding approaches.

A. What is the best acoustic word embedding model in the purely unsupervised case?

Table I shows the AP on development data for the unsupervised monolingual models from §III-C applied to the six zero-resource evaluation languages. All the neural models are trained on UTD pairs. For reference, the performance of supervised monolingual models trained using forced alignments and ground truth labels is also given. This can be seen as an upper bound where a perfect UTD system is available.

As in [18], we see that the unsupervised CAE-RNN outperforms downsampling and the AE-RNN on all six zero-resource languages. Here we additionally show that it also outperforms CLASSIFIERRNN and SIAMESERNN models when these are trained on UTD terms (the only exception is the SIAMESERNN on Mandarin achieving a higher AP of 28.4%). The CAE-RNN (UTD) even outperforms DTW on five of the languages, which is noteworthy since DTW has access to the full sequences for discriminating between words.

The supervised model variants (bottom section, Table I) still perform better than the unsupervised models (top section). In the supervised setting, the best performance is achieved by either the CAE-RNN or SIAMESERNN, depending on the language. This is different from the unsupervised setting where the CAE-RNN is better than the SIAMESERNN in most cases. One interpretation could be that, since the SIAMESERNN is trained in a discriminative fashion, it might be more susceptible

TABLE I
AP (%) ON DEVELOPMENT DATA FOR THE ZERO-RESOURCE LANGUAGES USING UNSUPERVISED ACOUSTIC WORD EMBEDDING MODELS. FOR REFERENCE, RESULTS FROM SUPERVISED MODEL VARIANTS (MAKING USE OF GROUND TRUTH WORD SEGMENTS AND LABELS) ARE ALSO GIVEN. THE BEST RESULTS OF UNSUPERVISED AND SUPERVISED MODELS ARE HIGHLIGHTED.

Model	ES	HA	HR	SV	TR	ZH
Unsupervised models:						
DTW	22.3	27.9	16.9	14.2	**22.2**	19.3
Downsampling	14.6	20.4	14.8	8.4	16.3	15.3
AE-RNN (UTD)	15.3	11.2	12.6	6.9	12.7	14.3
CAE-RNN (UTD)	**28.8**	**36.0**	**23.6**	**15.6**	20.4	20.8
CLASSIFIERRNN (UTD)	15.0	15.4	13.6	7.6	9.8	15.4
SIAMESERNN (UTD)	21.3	5.4	6.2	8.7	11.9	**28.4**
Supervised models:						
CAE-RNN (GT)	**71.9**	**77.8**	**76.0**	59.7	68.7	84.6
CLASSIFIERRNN (GT)	68.3	70.4	75.3	50.3	61.0	83.5
SIAMESERNN (GT)	69.3	73.3	73.1	**64.0**	**73.8**	**90.0**

TABLE II
AP (%) ON HAUSA DEVELOPMENT DATA WHEN SYSTEMATICALLY REDUCING THE NOISE FROM THE UTD DISCOVERY PROCEDURE.

Training set	CAE-RNN	SIAMESERNN
1. UTD terms	36.0	5.4
2. UTD with corrected end-pointing	29.8	7.7
3. UTD with corrected pair labels	44.5	37.7
4. UTD with corrected end-pointing and labels	69.7	60.7

TABLE III
AP (%) ON DEVELOPMENT DATA WHEN ADDING LANGUAGE CONDITIONING (LC) IN THE DECODER OF THE MULTILINGUAL CAE-RNN.

Multilingual model	ES	HA	HR	SV	TR	ZH
CAE-RNN	50.0	53.7	**41.3**	30.1	38.9	50.8
CAE-RNN-LC	**54.6**	**57.6**	40.9	**33.6**	**43.0**	**55.0**

TABLE IV
AP (%) ON TEST DATA FOR THE ZERO-RESOURCE LANGUAGES. THE UNSUPERVISED CAE-RNNS ARE TRAINED SEPARATELY FOR EACH ZERO-RESOURCE LANGUAGE ON SEGMENTS FROM A UTD SYSTEM APPLIED TO UNLABELLED MONOLINGUAL DATA. THE MULTILINGUAL MODELS ARE TRAINED ON GROUND TRUTH WORD SEGMENTS OBTAINED BY POOLING LABELLED TRAINING DATA FROM SIX WELL-RESOURCED LANGUAGES. THE BEST OVERALL RESULT ON EACH LANGUAGE IS HIGHLIGHTED.

Model	ES	HA	HR	SV	TR	ZH
Unsupervised models:						
DTW	36.2	23.8	17.0	27.8	16.2	35.9
Downsampling	24.0	15.9	14.2	18.9	11.0	26.6
CAE-RNN (UTD)	49.7	27.8	26.5	28.7	16.0	33.3
Multilingual models:						
CAE-RNN-LC	**72.8**	47.3	42.9	**46.4**	**35.1**	51.6
CLASSIFIERRNN	68.8	**47.7**	**43.5**	45.1	35.0	**54.5**
SIAMESERNN	65.8	43.7	39.3	44.6	28.2	46.1

to erroneous matches by the UTD system; the CAE-RNN, in contrast, is trained using a softer reconstruction-like loss which could be more robust to UTD errors.

To support this claim, Table II presents results on Hausa, where we systematically reduce the noise from the UTD system. Using forced alignments, we compare each UTD term with the ground truth word token with which it overlaps most. We first correct end-pointing: the start and end frame positions predicted by the UTD system are changed to match that of the ground truth word with maximal overlap (row 2). We then keep the UTD end-pointing, but only train on pairs where the ground truth labels match (row 3). In row 4, we correct both the end-pointing and only train on pairs with the same true label. We see that as the UTD pairs are updated with correct labels (row 3) and additionally also end-pointing (row 4), the difference in performance between the CAE-RNN and SIAMESERNN shrinks to around 9% absolute in AP.

B. The effect of language conditioning

In §III-B we described model variants of the multilingual CLASSIFIERRNN and CAE-RNN where the decoder of a model has access to language-specific information. To see if this leads to better generalisation to unseen languages, we compare model variants with and without language conditioning. In the CLASSIFIERRNN, we found that split prediction branches give slightly worse performance than using a single shared softmax layer. In contrast, Table III shows that the multilingual CAE-RNN improves slightly on the development data of most languages when a language embedding is added at each decoder time-step. (This improvement was also obtained on the German development data.) In the tables below we also denote the CAE-RNN with language conditioning as CAE-RNN-LC.

C. Does transfer from a multilingual model produce better embeddings than unsupervised learning on the target language?

Based on the above experiments, we use the CAE-RNN (UTD) as our monolingual unsupervised baseline and we use the variant of the multilingual CAE-RNN with language conditioning. We now turn to our main question: is a multilingual acoustic word embedding model trained on labelled but external data (§III-A) superior to an unsupervised monolingual model trained on unlabelled in-domain data from the target zero-resource language?

Table IV shows the performance of monolingual unsupervised and supervised multilingual acoustic word embedding models applied to test data from the six zero-resource languages. We see that the multilingual models consistently outperform the unsupervised models across all six zero-resource languages. Of the three supervised multilingual models, the SIAMESERNN performs worst. The relative performance of the multilingual CAE-RNN and CLASSIFIERRNN models is not consistent over the six zero-resource evaluation languages, with one model working better on some languages while another works better on others. Nevertheless, all three multilingual models consistently outperform the best unsupervised monolingual model trained directly on the target languages, showing the benefit of incorporating data from well-resourced languages where labels are available.

Despite the improvements of the multilingual over the unsupervised approach, the multilingual models never outperforms the best monolingual supervised models of Table I.[3] The multilingual models are therefore still not reaching the upper bound from supervised monolingual training.

It is interesting to note that, of the monolingual supervised models (bottom section Table I), the SIAMESERNN is one of the best models, outperforming the monolingual supervised CAE-RNN on three languages. In contrast, the multilingual SIAMESERNN never performs as well as the multilingual CAE-RNN when applied to unseen zero-resource languages (Table IV). This could again be due to the discriminative nature of the SIAMESERNN: it is explicitly trained to discriminate between pairs of words from several well-resourced training languages. Even though it might succeed in this task, this

[3]The two tables here actually report performance on different sets (development and test data, respectively). But this statement is true: the multilingual models in Table IV are always worse than their supervised monolingual counterparts when applied to the same development data.

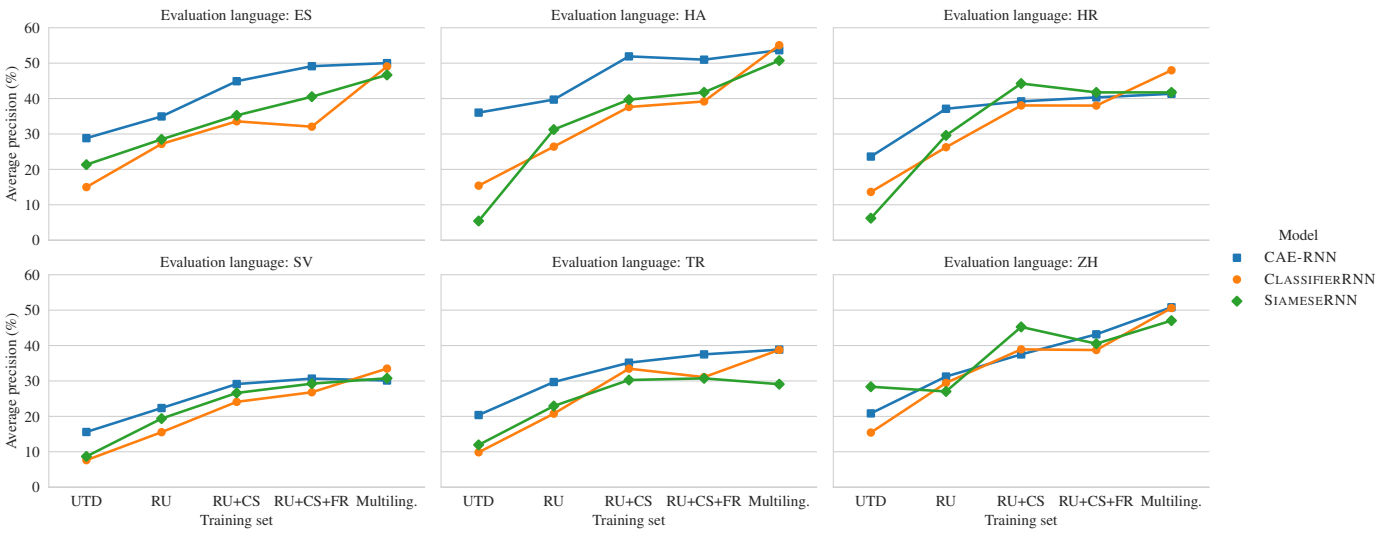

Fig. 4. AP (%) on development data for the six zero-resource language, as multilingual models are trained on one (RU), two (RU+CS), three (RU+CS+FR) and all six (multiling.) training languages. Scores from unsupervised models trained on UTD segments from the evaluation language are also given.

TABLE V
AP (%) ACHIEVED BY MULTILINGUAL MODELS WHEN APPLIED TO DEVELOPMENT DATA FROM THE SEEN TRAINING LANGUAGES.

Model	RU	CS	FR	PL	TH	PO
CAE-RNN-LC	**45.6**	62.6	46.9	45.3	80.8	64.7
SIAMESERNN	44.2	**76.7**	**47.8**	**47.8**	**85.8**	**72.7**

does not necessarily lead to generalisation to unseen languages. Table V shows the results when the two models are compared on development data from languages seen during training. The results show that the multilingual SIAMESERNN outperforms the CAE-RNN on five out of six of the training languages. This does not happen even once in Table IV when considering performance on unseen languages, which shows that the multilingual SIAMESERNN does not generalise as well.

VI. FURTHER ANALYSIS

In this analysis section we look at how the number and choice of training languages impact performance. We also perform probing experiments to obtain a better understanding of the resulting embedding spaces for the different models.

A. Are more languages beneficial for training multilingual embedding models?

Figure 4 shows development performance on each of the zero-resource language as multilingual models are trained on one (RU), two (RU+CS), three (RU+CS+FR) and all six (multiling.) well-resourced training languages. Here we use the multilingual CAE-RNN without language conditioning (§III-B). For reference, the results from monolingual unsupervised models are also given as the first point in each subplot. We can therefore interpret each plot as giving the results as we increase the number of training languages from zero (UTD) to six.

Results are not entirely consistent across the evaluation languages, but we can identify the following general trends. Firstly, using labelled data from a single training language (Russian) gives improvements over unsupervised embeddings for all six evaluation languages and all three model variants (except for the Russian SIAMESERNN applied to Mandarin). Secondly, when using fewer training languages, the multilingual CAE-RNN outperforms the multilingual CLASSIFIERRNN and SIAMESERNN in most cases. But, as we increase the number of training languages, all three models improve and it becomes less clear which is superior. On several languages the multilingual CLASSIFIERRNN gives similar or better performance than the CAE-RNN. Although the SIAMESERNN also improves as we add training languages, it is not the top-performing model on any of the six evaluation languages. Thirdly, the effect of additional training languages differs based on the evaluation language. Focusing only on the CAE-RNN, AP improves systematically as we add languages for Spanish, Turkish and Mandarin. But on Hausa, Croatian and Swedish, performance seems to plateau with two training languages (on Croatian this might not be surprising since both Russian and Czech are from the same language family).

Generally speaking, we can therefore conclude that adding more training languages does not deteriorate performance, but on some languages improvements are marginal. In all cases, however, using labelled training data from even a single well-resourced language improves performance over an unsupervised model trained only on the target language.

B. Is the training language choice important?

To investigate the impact of the particular choice of training language, we train supervised monolingual CAE-RNNs on each of the well-resourced languages and then apply these to each of the zero-resource languages. Results are given in Figure 5. The choice of well-resourced language can greatly impact performance. On Spanish, using Portuguese is better than any other language, and on Croatian the monolingual Czech system performs, showing the benefit of training on languages from the same family. The supervised Thai model

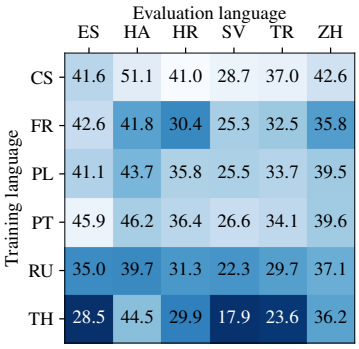

Fig. 5. AP (%) on development data for the six zero-resource languages (columns) when applying different monolingual CAE-RNN models, each trained on labelled data from a well-resourced language (rows). Heatmap colours are normalised for each zero-resource language (i.e. per column).

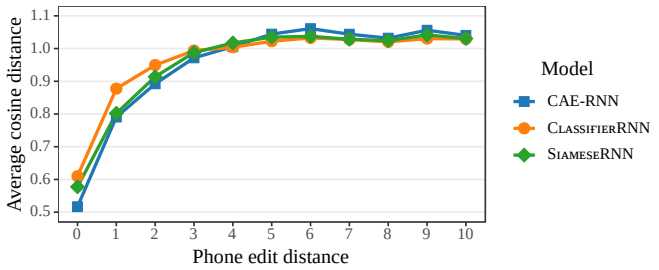

Fig. 6. Average cosine distances between pairs of embedded Hausa words as a function of their phone edit distance (development data, multilingual models).

TABLE VI
GOODNESS OF FIT (R^2, PROPORTION OF VARIANCE EXPLAINED) OF THE REGRESSION MODELS PREDICTING ABSOLUTE WORD DURATION ON DEVELOPMENT DATA FOR THE ZERO-RESOURCE LANGUAGES.

Model	ES	HA	HR	SV	TR	ZH
Unsupervised models:						
Downsampling	0.47	0.72	0.57	0.36	0.51	0.69
Multilingual models:						
CAE-RNN	**0.94**	**0.96**	**0.96**	**0.95**	**0.95**	**0.94**
ClassifierRNN	0.79	0.76	0.81	0.78	0.71	0.71
SiameseRNN	0.82	0.78	0.84	0.81	0.75	0.76

performs worst on all the Indo-European languages but not on Hausa and Mandarin.

Although performance can differ dramatically based on the source-target language pair, all of the systems in Figure 5 outperform the unsupervised monolingual CAE-RNNs trained on UTD pairs from the target language (Table I), except for the Thai-Spanish pair. Thus, training on just a single well-resourced language (even if it is not in the same language family) is beneficial. Furthermore, although adding languages does not always lead to improvements in Figure 4, the multilingual CAE-RNN trained on all six languages (Table IV, Figure 4) outperforms all of the supervised monolingual models in Figure 5 on the evaluation languages. The performance effects of training language choice therefore diminish as more training languages are used.

C. How are the embedding spaces organised?

In previous sections we compared models using the same-different task. In some cases there was little to separate the models, e.g., there are only small differences between the supervised multilingual CAE-RNN, ClassifierRNN and SiameseRNN when training on all six languages (right-most points, Figure 4). Rather than using the same-different task, here we try to directly characterise the organisation of the models' embedding spaces. This is valuable, not only for the sake of the analysis itself, but also from a technological perspective: it can help us anticipate potential failure modes when using the embeddings in downstream speech systems. As in the analysis of acoustic word embeddings in [24], we use a number of tests to probe the embedding spaces. The focus in [24] was on monolingual models, while here we focus on the multilingual embedding models (§III-A). Instead of reporting the findings from all the tests we performed, we only report those that showed some differences between the multilingual models.

We first consider whether the embedding spaces encode the *degree* of similarity between words. High AP in the same-different task indicates that instances of the same word are closer to each other in an embedding space than different words. But this does not tell us whether words that are similar but not identical are close to each other. To measure this, we consider the distance between pairs of embedded words as a function of the phone edit distance between them. The result for Hausa is shown in Figure 6. For all three multilingual embedding models, the average cosine distance between word pairs is lowest when the words are identical (a phone edit distance of 0), and this distance systematically increases as the words become more phonetically different. This effect seems strongest for the CAE-RNN, which has the lowest embedding distance when words are identical, but the largest distance when words differ by 5 or more edits. This suggests that embedding spaces are organised according to the words' phonetic similarity, and that this is more so in the multilingual CAE-RNN.

Next, we consider whether the embedding spaces capture information about word duration and speaker identity. To do this, we train and test linear classifiers (multi-class logistic regression models) and linear regression models to, respectively, predict the speaker identity of an embedded test word and its duration in milliseconds. In each case, we train a model on 80% of the embedded words from the development data of a language and then test the model on the held-out 20%.

We first consider word duration prediction, looking at the goodness of fit (R^2) of the corresponding linear regression models. Table VI shows that, while all three multilingual models perform well on this task, the CAE-RNN performs best on all six languages. Note that better performance on this task does not necessarily imply better word discrimination performance; this could depend on the language and the specific downstream setting under consideration. Nevertheless, this analysis shows that the CAE-RNN seems to capture differences in duration more than the other models (at this salient level measured by linear regression).

We next consider whether the embeddings allow for accurate speaker classification. Table VII shows that embeddings from

TABLE VII
SPEAKER CLASSIFICATION ACCURACY (%) ON DEVELOPMENT DATA FOR THE ZERO-RESOURCE LANGUAGES. THE HIGHEST SCORES ARE HIGHLIGHTED IN RED (UNDERLINE) AND THE LOWEST IN BLUE (ITALICS).

Model	ES	HA	HR	SV	TR	ZH
Unsupervised models:						
Downsampling	31.0	43.8	38.3	30.9	29.8	53.0
Multilingual models:						
CAE-RNN	34.3	38.7	33.9	36.4	36.4	42.1
CLASSIFIERRNN	28.3	28.5	27.9	*24.9*	30.4	32.0
SIAMESERNN	*25.7*	*25.4*	*26.4*	27.3	*26.4*	*31.0*

the multilingual CAE-RNN allow for the best speaker classification results of the three multilingual models. The SIAMESERNN generally performs the worst. This indicates that, at a level that can be captured by a linear classifier, the multilingual SIAMESERNN's embeddings are more speaker-invariant then those of the CAE-RNN. This is surprising since the CAE-RNN generally performs better in cross-speaker word discrimination (Table IV, Figure 4). However, being able to classify speaker using a linear projection of the embeddings does not necessarily imply worse word discrimination performance. Furthermore, for some downstream tasks it could actually be beneficial to retain some speaker information in the embeddings.

Finally, we are interested to see whether the multilingual embedding models separate the embedding spaces according to language. Again, being able to partition the embedding space according to language does not necessarily imply better or worse word discrimination performance, but it can be useful to know whether this happens for settings where words from multiple languages might be embedded at the same time. Moreover, if the models did perfectly separate the embedding spaces according to training language, the models would presumably not transfer to unseen languages. For the analysis, we embed at the same time triphones from five languages: RU, CS, FR, DE and ZH. Three of these languages are seen training languages, while two are unseen. This allows us to determine if the embeddings encode language identity information better for seen languages. A set of 25 common triphones are used, e.g., [ana] and [tam]. To ensure that the triphones are genuinely identical across languages, we unified the existing GlobalPhone transcriptions for these languages by converting them into the International Phonetic Alphabet format. As before, we apply a linear classifier on the models' embeddings.

Table VIII shows that the multilingual CAE-RNN's embeddings allows for the best language identification. Adding more training languages generally improves language classification performance, but this effect is stronger for the CLASSIFIERRNN and SIAMESERNN than for the CAE-RNN (which actually performs best when only trained on three languages). An analysis of the individual $F1$-scores (not shown here) per language shows there are no consistent differences between seen (RU, CS, FR) and unseen (DE, ZH) languages. Figure 7 shows a visualisation of the embedded instances of the [tat] triphone in all five languages for the multilingual CAE-RNN. While triphones from the same language seem to cluster together, there

TABLE VIII
ACCURACY (%) OF LINEAR CLASSIFIERS PREDICTING THE LANGUAGE IDENTITY OF TRIPHONES FROM THE DEVELOPMENT DATA OF FIVE LANGUAGES (THREE SEEN, TWO UNSEEN).

Model	RU	RU+CS	RU+CS+FR	Multilingual
CAE-RNN	**61.8**	**63.7**	**66.8**	**65.2**
CLASSIFIERRNN	54.3	54.2	56.2	61.8
SIAMESERNN	57.4	60.2	57.3	65.0

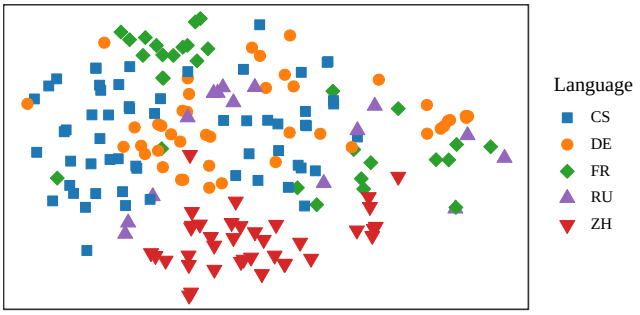

Fig. 7. Two-dimensional t-SNE plots of the triphone [tat] in five languages embedded using the multilingual CAE-RNN.

is a lot of overlap across languages, potentially enabling the model to generalise the acquired phonetic properties of training languages to unseen languages. Note, however, that such generalisation may not work equally well for all languages: in the figure there seems to be more overlap between the training languages (RU, CS, FR) and DE (another Indo-European language) than between the training languages and ZH (a Sino-Tibetan language).

To summarise, our probing experiments suggest that the multilingual CAE-RNN's embeddings encode more information about phonetic content, word duration, language identity, and speaker identity compared to the other multilingual models.

VII. CONCLUSION

We proposed multilingual transfer as a way to obtain acoustic word embeddings for zero-resource languages. We applied multilingual models trained jointly on six well-resourced languages to six zero-resource languages without any labelled data. Three multilingual models—a Siamese recurrent neural network (RNN), a classifier RNN and a correspondence autoencoder (CAE) RNN with a reconstruction-like loss—consistently outperformed monolingual unsupervised models trained directly on the zero-resource language. When fewer training languages are used, we showed that the multilingual CAE outperforms the other models and that performance is affected by the combination of training and test languages. These effects diminish as more training languages are used. Since there was little to distinguish between the three multilingual models, we performed further analysis to determine the types of properties captured by the different models. This revealed that the CAE is more sensitive than the other models to phonetic content, word duration information and language identity, but also (surprisingly) speaker identity. Future work will look to apply

these models in downstream systems and to consider more low-resource languages.

REFERENCES

[1] A. Jansen, E. Dupoux, S. J. 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.-y. 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 *Proc. ICASSP*, 2013.

[2] M. Versteegh, X. Anguera, A. Jansen, and E. Dupoux, "The Zero Resource Speech Challenge 2015: Proposed approaches and results," in *Proc. SLTU*, 2016.

[3] E. Dunbar, X. N. Cao, J. Benjumea, J. Karadayi, M. Bernard, L. Besacier, X. Anguera, and E. Dupoux, "The Zero Resource Speech Challenge 2017," in *Proc. ASRU*, 2017.

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

[5] T. J. Hazen, W. Shen, and C. White, "Query-by-example spoken term detection using phonetic posteriorgram templates," in *Proc. ASRU*, 2009.

[6] Y. Zhang and J. R. Glass, "Unsupervised spoken keyword spotting via segmental DTW on Gaussian posteriorgrams," in *Proc. ASRU*, 2009.

[7] K. Levin, A. Jansen, and B. Van Durme, "Segmental acoustic indexing for zero resource keyword search," in *Proc. ICASSP*, 2015.

[8] M. Elsner and C. Shain, "Speech segmentation with a neural encoder model of working memory," in *Proc. EMNLP*, 2017.

[9] C.-y. Lee, T. O'Donnell, and J. R. Glass, "Unsupervised lexicon discovery from acoustic input," *Trans. ACL*, vol. 3, pp. 389–403, 2015.

[10] O. J. Räsänen, G. Doyle, and M. C. Frank, "Unsupervised word discovery from speech using automatic segmentation into syllable-like units," in *Proc. Interspeech*, 2015.

[11] H. Kamper, K. Livescu, and S. Goldwater, "An embedded segmental k-means model for unsupervised segmentation and clustering of speech," in *Proc. ASRU*, 2017.

[12] L. R. Rabiner, A. E. Rosenberg, and S. E. Levinson, "Considerations in dynamic time warping algorithms for discrete word recognition," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 26, no. 6, pp. 575–582, 1978.

[13] K. Levin, K. Henry, A. Jansen, and K. Livescu, "Fixed-dimensional acoustic embeddings of variable-length segments in low-resource settings," in *Proc. ASRU*, 2013.

[14] Y.-A. Chung, C.-C. Wu, C.-H. Shen, and H.-Y. Lee, "Unsupervised learning of audio segment representations using sequence-to-sequence recurrent neural networks," *in Proc. Interspeech*, 2016.

[15] H. Kamper, W. Wang, and K. Livescu, "Deep convolutional acoustic word embeddings using word-pair side information," in *Proc. ICASSP*, 2016.

[16] S. Settle, K. Levin, H. Kamper, and K. Livescu, "Query-by-example search with discriminative neural acoustic word embeddings," in *Proc. Interspeech*, 2017.

[17] K. Audhkhasi, A. Rosenberg, A. Sethy, B. Ramabhadran, and B. Kingsbury, "End-to-end ASR-free keyword search from speech," *IEEE J. Sel. Topics Signal Process.*, vol. 11, no. 8, pp. 1351–1359, 2017.

[18] H. Kamper, "Truly unsupervised acoustic word embeddings using weak top-down constraints in encoder-decoder models," in *Proc. ICASSP*, 2019.

[19] S. Settle and K. Livescu, "Discriminative acoustic word embeddings: Recurrent neural network-based approaches," in *Proc. SLT*, 2016.

[20] H. Kamper, A. Anastassiou, and K. Livescu, "Semantic query-by-example speech search using visual grounding," in *Proc. ICASSP*, 2019.

[21] T. Schultz, N. T. Vu, and T. Schlippe, "GlobalPhone: A multilingual text & speech database in 20 languages," in *Proc. ICASSP*, 2013.

[22] S.-F. Huang, Y.-C. Chen, H.-y. Lee, and L.-s. Lee, "Improved audio embeddings by adjacency-based clustering with applications in spoken term detection," *arXiv preprint arXiv:1811.02775*, 2018.

[23] H. Kamper, Y. Matusevych, and S. J. Goldwater, "Multilingual acoustic word embedding models for processing zero-resource languages," in *Proc. ICASSP*, 2020.

[24] Y. Matusevych, H. Kamper, and S. Goldwater, "Analyzing autoencoder-based acoustic word embeddings," in *BAICS Workshop ICLR*, 2020.

[25] Y.-H. Wang, H.-y. Lee, and L.-s. Lee, "Segmental audio word2vec: Representing utterances as sequences of vectors with applications in spoken term detection," in *Proc. ICASSP*, 2018.

[26] Y. Yuan, C.-C. Leung, L. Xie, H. Chen, B. Ma, and H. Li, "Learning acoustic word embeddings with temporal context for query-by-example speech search," in *Proc. Interspeech*, 2018.

[27] Y.-C. Chen, S.-F. Huang, C.-H. Shen, H.-y. Lee, and L.-s. Lee, "Phonetic-and-semantic embedding of spoken words with applications in spoken content retrieval," in *Proc. SLT*, 2018.

[28] 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," *in Proc. Interspeech*, 2018.

[29] A. Haque, M. Guo, P. Verma, and L. Fei-Fei, "Audio-linguistic embeddings for spoken sentences," in *Proc. ICASSP*, 2019.

[30] Y. Shi, Q. Huang, and T. Hain, "Contextual joint factor acoustic embeddings," *arXiv preprint arXiv:1910.07601*, 2019.

[31] S. Palaskar, V. Raunak, and F. Metze, "Learned in speech recognition: Contextual acoustic word embeddings," in *Proc. ICASSP*, 2019.

[32] Y.-A. Chung and J. R. Glass, "Speech2vec: A sequence-to-sequence framework for learning word embeddings from speech," in *Proc. Interspeech*, 2018.

[33] S. Bengio and G. Heigold, "Word embeddings for speech recognition," in *Proc. Interspeech*, 2014.

[34] S. Ghannay, Y. Estève, N. Camelin, and P. Deléglise, "Evaluation of acoustic word embeddings," in *Proc. ACL Workshop Evaluating Vector-Space Representations NLP*, 2016, pp. 62–66.

[35] W. He, W. Wang, and K. Livescu, "Multi-view recurrent neural acoustic word embeddings," in *Proc. ICLR*, 2017.

[36] S. Settle, K. Audhkhasi, K. Livescu, and M. Picheny, "Acoustically grounded word embeddings for improved acoustics-to-word speech recognition," in *Proc. ICASSP*. IEEE, 2019, pp. 5641–5645.

[37] M. Jung, H. Lim, J. Goo, Y. Jung, and H. Kim, "Additional shared decoder on Siamese multi-view encoders for learning acoustic word embeddings," in *Proc. ASRU*, 2019.

[38] K. Veselỳ, M. Karafiát, F. Grézl, M. Janda, and E. Egorova, "The language-independent bottleneck features," in *Proc. SLT*, 2012.

[39] Y. Yuan, C.-C. Leung, L. Xie, H. Chen, B. Ma, and H. Li, "Pairwise learning using multi-lingual bottleneck features for low-resource query-by-example spoken term detection," in *Proc. ICASSP*, 2017.

[40] E. Hermann, H. Kamper, and S. J. Goldwater, "Multilingual and unsupervised subword modeling for zero-resource languages," *arXiv preprint arXiv:1811.04791*, 2018.

[41] R. Menon, H. Kamper, E. Van Der Westhuizen, J. Quinn, and T. R. Niesler, "Feature exploration for almost zero-resource asr-free keyword spotting using a multilingual bottleneck extractor and correspondence autoencoders," in *Proc. Interspeech*, 2019.

[42] L. Ondel, H. K. Vydana, L. Burget, and J. Černockỳ, "Bayesian subspace hidden markov model for acoustic unit discovery," *arXiv preprint arXiv:1904.03876*, 2019.

[43] T. Schultz and A. Waibel, "Language-independent and language-adaptive acoustic modeling for speech recognition," *Speech Communication*, vol. 35, pp. 31–51, 2001.

[44] T. R. Niesler, "Language-dependent state clustering for multilingual acoustic modelling," *Speech Commun.*, vol. 49, no. 6, pp. 453–463, 2007.

[45] S. Toshniwal, T. N. Sainath, R. J. Weiss, B. Li, P. Moreno, E. Weinstein, and K. Rao, "Multilingual speech recognition with a single end-to-end model," in *Proc. ICASSP*, 2018.

[46] S. Tong, P. N. Garner, and H. Bourlard, "Multilingual training and cross-lingual adaptation on ctc-based acoustic model," *Speech Commun.*, vol. 104, pp. 39–46, 2018.

[47] J. Cho, M. K. Baskar, R. Li, M. Wiesner, S. H. Mallidi, N. Yalta, M. Karafiat, S. Watanabe, and T. Hori, "Multilingual sequence-to-sequence speech recognition: architecture, transfer learning, and language modeling," in *Proc. SLT*, 2018.

[48] O. Adams, M. Wiesner, S. Watanabe, and D. Yarowsky, "Massively multilingual adversarial speech recognition," in *Proc. ACL*, 2019.

[49] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Trans. Knowl. Data Eng.*, vol. 22, no. 10, pp. 1345–1359, 2009.

[50] S. Ruder, "Neural transfer learning for natural language processing," Ph.D. dissertation, NUI Galway, Ireland, 2019.

[51] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[52] J. Bromley, J. W. Bentz, L. Bottou, I. Guyon, Y. LeCun, C. Moore, E. Säckinger, and R. Shah, "Signature verification using a "Siamese" time delay neural network," *Int. J. Pattern Rec.*, vol. 7, no. 4, pp. 669–688, 1993.

[53] K. Q. Weinberger and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," *J. Mach. Learn. Res.*, vol. 10, no. Feb, pp. 207–244, 2009.
[54] G. Chechik, V. Sharma, U. Shalit, and S. Bengio, "Large scale online learning of image similarity through ranking," *J. Mach. Learn. Res.*, vol. 11, pp. 1109–1135, 2010.
[55] F. Schroff, D. Kalenichenko, and J. Philbin, "FaceNet: a unified embedding for face recognition and clustering," in *Proc. CVPR*, 2015.
[56] A. Jansen and B. Van Durme, "Efficient spoken term discovery using randomized algorithms," in *Proc. ASRU*, 2011.
[57] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, 2015.
[58] M. A. Carlin, S. Thomas, A. Jansen, and H. Hermansky, "Rapid evaluation of speech representations for spoken term discovery," in *Proc. Interspeech*, 2011.
[59] E. Hermann, H. Kamper, and S. J. Goldwater, "Multilingual and unsupervised subword modeling for zero resource languages," *Comput. Speech Language*, 2020.