Linguistically-informed Training of Acoustic Word Embeddings for Low-resource Languages

Zixiaofan Yang, Julia Hirschberg

Columbia University, USA

zy2231@columbia.edu, julia@cs.columbia.edu

Abstract

Acoustic word embeddings have been proven to be useful in query-by-example keyword search. Such embeddings are typically trained to distinguish the same word from a different word using exact orthographic representations; so, two different words will have dissimilar embeddings even if they are pronounced similarly or share the same stem. However, in real-world applications such as keyword search in low-resource languages, models are expected to find all derived and inflected forms for a certain keyword. In this paper, we address this mismatch by incorporating linguistic information when training neural acoustic word embeddings. We propose two linguistically-informed methods for training these embeddings, both of which, when we use metrics that consider non-exact matches, outperform state-of-the-art models on the Switchboard dataset. We also present results on Sinhala to show that models trained on English can be directly transferred to embed spoken words in a very different language with high accuracy.

Index Terms: acoustic word embeddings, Siamese neural networks, low-resource languages

1. Introduction

There are more than 7,000 languages in the world, but human language technology, especially speech processing technology, exists only for a few of them. Building an automated speech processing system for a new language requires extensive time and effort to collect, transcribe, and translate data. However, a crucial real-world application for low-resource language speech processing systems is to support a rapid and effective response to emerging incidents, given limited language resources. In this scenario, rather than building automatic speech recognition and machine translation systems with insufficient accuracy, it may be more helpful to provide situational awareness by simply identifying critical elements of information, such as incident-related topics, names, and events. This corresponds to query-by-example keyword search, where spoken keyword queries are used to retrieve information from speech utterances. In the work presented here, our goal is to obtain acoustic word embeddings that can be easily applied to query-by-example keyword search in languages with minimal resources.

The idea of acoustic word embeddings is to represent variable-length speech segments with fixed-dimensional feature vectors [1]. With the advance in neural networks, researchers [2, 3, 4, 5] have shown that neural acoustic word embeddings trained on word-pair information can better discriminate words than dynamic time warping (DTW) based approaches. A common method for obtaining acoustic word embeddings is to train a Siamese neural network [6] which has parameter-sharing components that take in pairs of spoken word as inputs and minimize or maximize the output distance depending on whether the pair comes from the same or different word class. All existing works use exact orthographic representation to distinguish whether two speech samples are from the same or different words, no matter how close or far they actually are. For example, a model trained in such a method will learn to maximize the embedding distance of a spoken sample of "terrorist" and a sample of "terrorism", even if they share the same stem and are phonetically similar. However, in real-world applications, keyword search models are expected to find all the derived and inflected forms for a certain keyword, which means that these embeddings should be as close to one another as possible. To address the mismatch between the standard training and this different use of acoustic word embeddings, we propose two linguistically-informed methods: (1) Group words by stems during training and treat words with the same stem as being from the same word class. (2) Instead of minimizing or maximizing output distance for the same or different word pairs, calculate the pronunciation distance of each word pair and try to make the output distance equal to the pronunciation distance. We evaluate these methods on two different experimental settings: (1) a low-resource setting, where training data is available but limited; (2) a zero-resource setting, where no training data is available for the target language.

The remainder of the paper is organized as follows. In Section 2, we describe related work on keyword search in low-resource languages and acoustic word embeddings. We introduce our linguistically-informed training methods for neural acoustic word embeddings in Section 3. Experimental settings and results are presented in Sections 4 and 5. We conclude in Section 6 and present directions for future work.

2. Related work

For keyword search in low-resource languages, Jansen and Durme [7] mapped frame-level feature vectors to a sortable-bit signature and used DTW-inspired methods to identify frame-level matches. Levin et al. [8] extended previous work to embed speech segments in an unsupervised way and performed search directly at the segment level. Chen et al. [9] built a keyword search system using word-morph interpolated language models to mitigate the data sparsity issue of the morphologically-rich vocabulary in Tamil. Another work of Chen et al. [10] added phonological and prosodic features for keyword search in Swahili, including the duration, speaking rate, and number of vowels and consonants of keyword queries. In the 2014 Query-by-Example Speech Search Task (QUESST) [11], one task was non-exact matching, in which test occurrences could contain small morphological variations with regard to the lexical form of the query. To solve this problem, Xu et al. [12] proposed a partial matching strategy in which all partial phone sequences of a query were used to search for matching instances; Proenga et al. [13] modified the DTW algorithm using posteriorgrams and extracted intricate paths to account for special cases. Although some of these works attempted to solve the non-exact matching problem, they all used DTW-based matching on frame-level representations, which has been shown to be outperformed by distance-based matching on acoustic word embeddings [2, 3, 4].

Most research on acoustic word embeddings has used Siamese neural networks with orthographic word-pair information. Kamper et al. [2] built such models using convolutional neural networks (CNNs), and Settle and Livescu [3] used recurrent neural networks (RNNs) to learn embeddings. Settle et al. [4] also found that these embeddings could be used for query-by-example search with substantial improvements in performance over DTW-based approaches. Yuan et al. [5] used temporal context padding instead of zero padding on each training word to better match the sliding window with context at test time. Yuan et al. [14] trained a bottleneck feature (BNF) extractor and fed BNFs instead of cepstral features into Siamese networks. Gundogdu and Saraclar [15, 16] learned frame-level distance metrics automatically for similarity measurement.

For autoencoder-based unsupervised acoustic word embeddings, Chung at al. [17] and Holzenberger et al. [18] used a recurrent autoencoder to learn embeddings without using word-pair information as supervision. However, both research efforts assumed that training utterances are already segmented into words while learning embeddings in an unsupervised way. Kamper [19] solved this mismatch by using an unsupervised term discovery system to find sample same-word pairs. For evaluating acoustic word embeddings, Ghannay et al. [20, 21] proposed to evaluate the intrinsic performances of acoustic word embeddings by comparing embedding similarity with the orthographic and phonetic similarity of the original words. There is also work on embedding phonetic information and orthographic representation jointly [22, 23]. However, these methods require a considerable amount of speech data with word-aligned transcripts in the target language which may not be possible in a low-resource setting.

3. Our approaches

In this section, we describe our approach to acoustic word embeddings using Siamese neural networks and two linguistically-informed training methods. In the first method, we modify the definition of word class in traditional methods by treating words with the same stem as belonging to the same word class. In the second method, we propose a new loss function to directly train acoustic embeddings that retain the phonetic distance of the orthographic representations.

3.1. Siamese neural network with triplet loss function

A Siamese neural network usually consists of two or three parameter-sharing components which accept distinct inputs but are joined by a loss function at the top. The loss function is used to compute a metric between the highest-level feature representation of each component. One commonly used loss function for learning embeddings is triplet loss, which accepts a triplet of embedding vectors from the three components. Each triplet consists of three embedded samples (x_a, x_p, x_n), and the loss over the embeddings is defined as:

$$Loss(x_a, x_p, x_n) = max\{0, m + d(x_a, x_p) - d(x_a, x_n)\} \quad (1)$$

where x_a is the embedding of the current sample (anchor), x_p is the embedding of a positive sample from the same word class, and x_n is the embedding of a negative sample from a different word class. By optimizing triplet loss with margin m, the embeddings of samples in the same word class will be close to each other, while the embeddings of samples in different word classes will be far apart. The distance metric $d(x_1, x_2)$ can be any function mapping two same-dimensional vectors to a distance score. Figure 1 shows a Siamese neural network with triplet loss, where the inputs are processed by the components from bottom to top and joined by the triplet loss function.

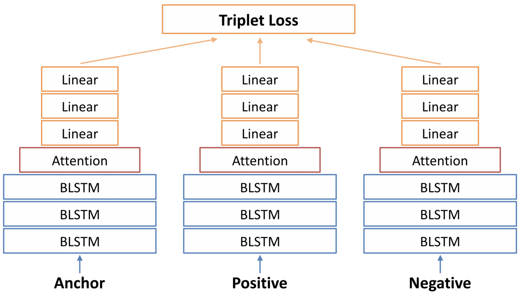

Figure 1: *Siamese neural network with triplet loss.*

3.2. Linguistically-informed training methods

Clustering words by their stems. As described in Section 1, keyword search models in real applications are often expected to find morphological variants of keywords, which means these embeddings should be as close as possible. In this method, we address this by clustering all words with the same stem into a single class. We use the same triplet loss function as in Equation 1 but with a different strategy for retrieving triplets. For each anchor sample, a positive sample is another word with the same stem, and a negative sample is drawn from words with different stems. By training the Siamese neural network on stems, we expect the model to focus on the stems of the words instead of the full words.

Learning pronunciation distance. This method is motivated by another limitation of a triplet loss function using word-pair information. For each anchor, the model treats all negative samples similarly and attempts to make all the embeddings of negative samples far from the anchor's embedding. However, for homophones or other words pronounced similarly, we do not want these embeddings to be far apart. Thus our goal here is to obtain acoustic embeddings that retain the phonetic distance over word pairs. The loss is defined as

$$Loss(x_1, x_2) = (d(x_1, x_2) - d_{edit}(phone_1, phone_2))^2 \quad (2)$$

where x_1 and x_2 are the embeddings of two randomly chosen words, d can be any distance metric, and d_{edit} is the phonetic edit distance of the two words. By minimizing the mean squared error (MSE) between the embedding distance and the pronunciation distance of all word pairs, these embeddings can capture phonetic information more accurately.

4. Experiments

We conducted experiments in both low-resource and zero-resource settings using our baseline Siamese network model and the two proposed models described above. We evaluated the models on various metrics, measuring different characteristics of the embeddings.

4.1. Low-resource setting: English Switchboard

To simulate a low-resource setting, we performed experiments on a very limited subset of the Switchboard [24] corpus. We used the same train/dev/test file partitions as Settle et al. [4] and segmented the conversation into words according to the word-aligned transcripts. Following existing research [3, 4], we used only words with 0.5s to 2.0s duration and with a minimum of 6 characters. In order to use triplet loss function, the minimum occurrence of each word was set at 2 for the training set and 0 for the dev and test sets. After preprocessing, we obtained 10k, 11k and 11k samples on the train, dev, and test set respectively. There is less than 2 hours of speech for training, which corresponds to a low-resource setting. To use our proposed methods, we assumed that a stemmer and a dictionary with pronunciation information were available for the training language. In our experiments, we used the NLTK Snowball stemmer [25] and the CMU Pronouncing Dictionary for English.

4.2. Zero-resource setting: Sinhala

In a zero-resource setting, we assume that there is no transcribed speech data available in the target language, but that we can access native speakers of the target language for a limited time to obtain spoken keywords. We used Sinhala, one of the official languages in Sri Lanka. Sinhala is a morphologically rich language belonging to the Indo-European language family and is written using the Sinhalese script. We collected spoken keywords in Sinhala as part of the 2018 DARPA LORELEI (Low Resource Languages for Emergent Incidents) Evaluation [26]. Our goal was to identify *Situation Frames*, which are incidents that first responders are asked to help with, often in countries where they do not know the language, but will have an urgent need to know where aid is most needed and what kind of aid is required. Their main sources will be text and speech resources in the target language, such as social media posts and news broadcasts. Since there may be many of these in an emergency situation, they need to screen the data to discover critical information.

To tackle this low-resource language problem, we followed this process: first, we produced a list of 75 English keywords related to Situation Frames we were asked to identify, such as "medicine" for "Medical Assistance Frame", "terrorist" for "Terrorism or other Extreme Violence Frame", and "collapse" for "Infrastructure Frame". We were given online access for 2 hours to non-linguist native speakers of Sinhala, in which we asked them to translate each English keyword into 1 to 3 different words in the target language, which we then recorded. After this 2-hour meeting, we manually segmented the recorded words, obtaining 121 unique incident-related Sinhalese words and 610 spoken samples. All speech segments were then converted to single-channel audio with 8kHz sampling rate.

For a potential zero-resource setting, we believe that transferring knowledge directly from other high-resource languages is the most efficient method. In this scenario, we trained models on the full Switchboard English dataset and tested on the Sinhala spoken words. Since the training in this case did not use any Sinhala data, neither a Sinhalese stemmer nor a Sinhalese pronunciation dictionary was used.

4.3. Siamese neural network model details

We extracted Mel Frequency Cepstral Coefficients (MFCCs) for each word to use as input to our Siamese neural network models. The frame length for extracting MFCCs was 25ms and the stride was 10ms. For each frame, we computed a 39-dimensional feature vector composed of the first 13 MFCCs, their corresponding 13 deltas, and 13 delta-deltas. First, we used a set of recurrent layers to embed the input sequences while taking the frame-level temporal context into consideration. Recurrent layers are commonly used in learning acoustic embeddings, and RNN-based models have proven to be better than their CNN counterparts [3]. After the recurrent layers, an attention layer was used to focus on frames with potentially useful information and a set of fully connected layers was used to compute the final output. To our knowledge, this is the first work to use an attention mechanism in learning acoustic word embeddings. This mechanism helps our models focus on different parts of words dynamically and it suits our goal of learning robust embeddings with regard to small phonetic and morphological variations. Since the focus of our work is not to find the set of hyperparameters that performs the best on a certain dataset, we followed the optimal model configuration in Settle and Livescu [3] and Settle et al. [4]. For the recurrent layers, we used 3 bidirectional long short-term memory layers (BLSTMs) with 256 hidden cells in each direction and 0.3 dropout between layers. The fully connected layers have 1024 cells with Rectified linear unit (ReLU) non-linearities and 0.5 dropout between layers, and the final embeddings produced by the model have 1024 dimensions. We used the Adam optimization algorithm [27] with a learning rate of 0.001. The batch size was 32 and zero-padding was used. We used 1.0 as the margin of the triplet loss and Euclidean distance as similarity measurement. We randomly selected 5 negative samples for each anchor and only used the negative sample that most violated the marginal constraint to compute the triplet loss. For the low-resource setting experiments, we saved the checkpoint with the highest average precision on the dev set. For the zero-resource setting experiments, we trained all models for 500 epochs.

We trained three models for each setting using the same structure for the Siamese components in each model. "Word Triplet" is a baseline Siamese network optimized on triplet loss with word-pair information; "Stem Triplet" is the first proposed model optimized on triplet loss with stem-clustered word-pair information; "Pronunciation Dist" is the second model trained to learn the pronunciation distance of word pairs.

4.4. Evaluation metrics

Models were evaluated based on an acoustic word discrimination task, often used as a proxy for query-by-example keyword search [2, 3, 14, 22]. In this task, the models need to embed all test words and retrieve same-word pairs according to the distance between embeddings. Given a certain threshold, each word pair can be classified as the same or different if its distance is below or above the threshold. By varying the threshold, we can obtain a precision-recall curve and calculate average precision (Word AP) accordingly. However, in our scenarios, we are also interested in retrieving words with small morphological variation. To evaluate the performance of models on non-exact matching, we calculated another version of average precision (Stem AP) in which all words with the same stem are treated as coming from the same word class. In addition to precision-recall-curve-based metrics, we also used a phonetic-similarity based metric (Phonetic Sim) proposed by Ghannay et al. [20, 21] to evaluate whether embeddings capture phonetic similarity. In this metric, average embedding distances between each test word and other words are compared to the words' phonetic distances using Pearson correlation. For the evaluation in

Table 1: *Results on Switchboard test set.*

Model	Word AP	Stem AP	Phonetic Sim
Word Triplet	**44.5**	47.8	23.3
Stem Triplet	42.3	**54.1**	21.7
Pronunciation Dist	26.8	27.3	**38.8**

Table 2: *Results from training on English words, stems, and pronunciation distance but testing on Sinhala spoken keywords.*

Model	Word AP	Word P@4
Word Triplet	57.2	**81.6**
Stem Triplet	**60.3**	81.1
Pronunciation Dist	24.4	76.1

the zero-resource setting, since we did not have a stemmer or pronunciation dictionary for Sinhala, we used precision at *k* as an alternative evaluation metric. Each Sinhala word was spoken 5-6 times, so each sample has at least 4 other samples with the same word class and *k* is set at 4 for the metric (P@4).

5. Results and analysis

5.1. Low-resource setting: Switchboard

In Table 1 we present evaluation results on the Switchboard test set measured with word pair average precision (Word AP), average precision when clustering words by stems (Stem AP), and embedding phonetic similarity (Phonetic Sim). From these results, we see that the model trained using orthographic representations of words achieves the highest AP when the test words are represented by their full spellings; the model trained on word stems outperforms other models when test words are clustered by their stems; the model learning pronunciation distance best captures the phonetic information of words. We also note that the "Stem Triplet" model only performs 2.2 lower than the baseline model in Word AP, but is 6.3 higher in Stem AP. This means that the "Stem Triplet" model only sacrifices a small amount in exact word matching, but is better able to find words with some morphological variation. Moreover, although the "Pronunciation Dist" model is not as useful as other models at clustering same words together, it is significantly better at measuring the pronunciation distance of word pairs. These results suggest that we should choose the model to use depending on the problem we want to solve. Note that the best AP on the same train/dev/test set was achieved at 67.1 [3]. However, all conversations (59 hours) were used for data normalization in that work, which we cannot use in our low-resource scenario in which only 2 hours of training data are available. We match the work as much as possible by using the same neural network architecture and hyperparameters in all three models.

5.2. Zero-resource setting: Sinhala

For our zero-resource experiment on Sinhala, we used average precision (Word AP) and precision at 4 (Word P@4) as metrics. All models were trained on English Switchboard and tested on Sinhala spoken keywords without using any Sinhalese-specific adaptation. From Table 2, we observe that the model trained on triplet loss with English stems performs best on embedding Sinhala spoken words. All models achieve P@4 around 80 percents, meaning that we can retrieve other samples of the same word with high accuracy. Note that both AP and P@4 treat words with the same stem as different words, since we do not have an accurate stemmer in Sinhala. In order to check whether the embeddings can cluster words that are semantically and phonetically similar with each other, we visualize the embeddings of "Stem Triplet" model using t-Distributed Stochastic Neighbor Embedding (t-SNE) [28] as shown in Figure 2. Each word corresponds to 5 to 6 points in the figure, with different colors for different words. In order to demonstrate the clustering effect, we manually assigned similar colors to words with similar meanings. We used boxes to highlight some of the clusters with semantically and phonetically similar words: the green box contains Sinhala words for {destroy, destroyed, destruction}; the red box contains {attack, attacker}; the yellow box contains {crime, criminal, criminal (another translation in Sinhala)}; the blue box contains {hospital, hospitalize}; the pink box contains {terrorist, terrorism}. The boxes show that words with morphological variations can be embedded closely, which makes non-exact matching keyword search possible.

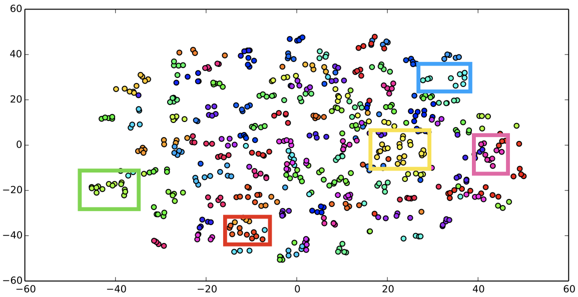

Figure 2: *t-SNE visualization of Sinhala spoken keywords.*

6. Conclusions

In this paper, we have proposed two linguistically-informed training methods for generating acoustic word embeddings that better suit real-world information retrieval applications. From the experimental results on Switchboard with a low-resource setting, we find that the traditional model trained with word-pair information is only useful for finding exact word matches, while the model trained with stem information can retrieve non-exact matches of words with small morphological variations, and the model learning pronunciation distance is best at measuring similarity between random words. We suggest that different training methods should be chosen depending on the purpose they are used for. The zero-source experiments in Sinhala demonstrate that we can generate embeddings that cluster similar words together without any training data in the target language. In future, we will explore whether the proposed training methods can be combined together to obtain embeddings that perform well in both clustering words and measuring distance. We will also perform query-by-example keyword search in low-resource and zero-resource settings with these methods.

7. Acknowledgment

This work was funded by DARPA LORELEI grant HR0011-15-2-0041. The views expressed in this paper however are those of the authors and do not reflect the official policy or position of the Department of Defense or the U.S government.

8. References

[1] K. Levin, K. Henry, A. Jansen, and K. Livescu, "Fixed-dimensional acoustic embeddings of variable-length segments in low-resource settings," in *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*. IEEE, 2013, pp. 410–415.

[2] H. Kamper, W. Wang, and K. Livescu, "Deep convolutional acoustic word embeddings using word-pair side information," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 4950–4954.

[3] S. Settle and K. Livescu, "Discriminative acoustic word embeddings: recurrent neural network-based approaches," in *2016 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 2016, pp. 503–510.

[4] S. Settle, K. Levin, H. Kamper, and K. Livescu, "Query-by-example search with discriminative neural acoustic word embeddings," *Proc. Interspeech 2017*, pp. 2874–2878, 2017.

[5] 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," *Proc. Interspeech 2018*, pp. 97–101, 2018.

[6] J. Bromley, I. Guyon, Y. LeCun, E. Säckinger, and R. Shah, "Signature verification using a "Siamese" time delay neural network," in *Advances in neural information processing systems*, 1994, pp. 737–744.

[7] A. Jansen and B. V. Durme, "Indexing raw acoustic features for scalable zero resource search," in *Thirteenth Annual Conference of the International Speech Communication Association*, 2012.

[8] 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)*. IEEE, 2015, pp. 5828–5832.

[9] N. F. Chen, C. Ni, I.-F. Chen, S. Sivadas, H. Xu, X. Xiao, T. S. Lau, S. J. Leow, B. P. Lim, C.-C. Leung *et al.*, "Low-resource keyword search strategies for Tamil," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 5366–5370.

[10] N. F. Chen, H. Xu, X. Xiao, C. Ni, I.-F. Chen, S. Sivadas, C.-H. Lee, E. S. Chng, B. Ma, H. Li *et al.*, "Exemplar-inspired strategies for low-resource spoken keyword search in Swahili," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2016, pp. 6040–6044.

[11] X. Anguera, L.-J. Rodriguez-Fuentes, A. Buzo, F. Metze, I. Szöke, and M. Penagarikano, "QUESST2014: Evaluating query-by-example speech search in a zero-resource setting with real-life queries," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 5833–5837.

[12] H. Xu, P. Yang, X. Xiao, L. Xie, C.-C. Leung, H. Chen, J. Yu, H. Lv, L. Wang, S. J. Leow *et al.*, "Language independent query-by-example spoken term detection using n-best phone sequences and partial matching," in *2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2015, pp. 5191–5195.

[13] J. Proenga, A. Veiga, and F. Perdigão, "Query by example search with segmented dynamic time warping for non-exact spoken queries," in *2015 23rd European Signal Processing Conference (EUSIPCO)*. IEEE, 2015, pp. 1661–1665.

[14] Y. Yuan, C.-C. Leung, L. Xie, B. Ma, and H. Li, "Learning neural network representations using cross-lingual bottleneck features with word-pair information," in *Interspeech*, 2016, pp. 788–792.

[15] B. Gündoğdu and M. Saraçlar, "Distance metric learning for posteriorgram based keyword search," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 5660–5664.

[16] B. Gündogdu and M. Saraclar, "Similarity learning based query modeling for keyword search," in *INTERSPEECH*, 2017, pp. 3617–3621.

[17] 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," *Interspeech 2016*, pp. 765–769, 2016.

[18] 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 *Interspeech 2018*. ISCA, 2018.

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

[20] S. Ghannay, Y. Esteve, N. Camelin, and P. Deléglise, "Evaluation of acoustic word embeddings," in *Proceedings of the 1st Workshop on Evaluating Vector-Space Representations for NLP*, 2016, pp. 62–66.

[21] S. Ghannay, Y. Estève, N. Camelin, and P. Deléglise, "Acoustic word embeddings for ASR error detection." in *INTERSPEECH*, 2016, pp. 1330–1334.

[22] W. He, W. Wang, and K. Livescu, "Multi-view recurrent neural acoustic word embeddings," *arXiv preprint arXiv:1611.04496*, 2016.

[23] 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," *arXiv preprint arXiv:1807.08089*, 2018.

[24] J. J. Godfrey, E. C. Holliman, and J. McDaniel, "Switchboard: telephone speech corpus for research and development," in *[Proceedings] ICASSP-92: 1992 IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 1, March 1992, pp. 517–520 vol.1.

[25] E. Loper and S. Bird, "NLTK: the natural language toolkit," *arXiv preprint cs/0205028*, 2002.

[26] S. M. Strassel and J. Tracey, "LORELEI language packs: Data, tools, and resources for technology development in low resource languages." in *LREC*, 2016.

[27] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *International Conference on Learning Representations*, 12 2014.

[28] L. v. d. Maaten and G. Hinton, "Visualizing data using t-SNE," *Journal of machine learning research*, vol. 9, no. Nov, pp. 2579–2605, 2008.