PHONEME BASED EMBEDDED SEGMENTAL K-MEANS FOR UNSUPERVISED TERM DISCOVERY

Saurabhchand Bhati[⋆]*, Herman Kamper*[†] *and K. Sri Rama Murty*[⋆]
ee12b1044@iith.ac.in, kamperh@gmail.com, ksrm@iith.ac.in

[⋆]Department of Electrical Engineering, IIT Hyderabad
[†]Electrical and Electronic Engineering, Stellenbosch University, South Africa

ABSTRACT

Identifying and grouping the frequently occurring word-like patterns from raw acoustic waveforms is an important task in the zero resource speech processing. Embedded segmental K-means (ES-KMeans) discovers both the word boundaries and the word types from raw data. Starting from an initial set of subword boundaries, the ES-Kmeans iteratively eliminates some of the boundaries to arrive at frequently occurring longer word patterns. Notice that the initial word boundaries will not be adjusted during the process. As a result, the performance of the ES-Kmeans critically depends on the initial subword boundaries. Originally, syllable boundaries were used to initialize ES-Kmeans. In this paper, we propose to use a phoneme segmentation method that produces boundaries closer to true boundaries for ES-KMeans initialization. The use of shorter units increases the number of initial boundaries which leads to a significant increment in the computational complexity. To reduce the computational cost, we extract compact lower dimensional embeddings from an auto-encoder. The proposed algorithm is benchmarked on Zero Resource 2017 challenge, which consists of 70 hours of unlabeled data across three languages, viz. English, French, and Mandarin. The proposed algorithm outperforms the baseline system without any language-specific parameter tuning.

Index Terms— Zero Resource speech processing, unsupervised learning, spoken term discovery, word segmentation

1. INTRODUCTION

Speech technologies rely on large corpora of transcribed speech audio data, pronunciation dictionaries and texts data for language modeling. Transcribing speech requires manual expertise and thus is very expensive and time-consuming. Zero resource speech technologies aim to develop unsupervised methods to discover the linguistic structure and lexicon directly from audio [1, 2]. These methods are crucial for extending speech technologies to the new languages with limited resources. Infants acquire their native languages in a largely unsupervised way [3] and developing speech technologies in zero resource settings may shed light upon speech inquisition process in children. Zero resource speech processing has been used for several applications including keyword spotting [4], unsupervised representation learning [5, 6, 7], topic discovery from untranscribed utterances [8], unsupervised acoustic unit modelling [7, 9, 10, 11, 12, 13] and language identification [14].

Unsupervised term discovery [7, 13, 15], which aims to find the repeatedly occurring word like patterns from the untranscribed audio data, is an important task in zero resource speech processing [1, 2]. Initial approaches [16] focused on finding isolated segments covering only a fraction of the speech data. Recent methods segment and cluster the entire speech data into word-like units [7, 17, 18, 19]. These full coverage system can be used to develop downstream applications like query-by-example search and speech indexing in a manner similar to when the supervised transcriptions are available [20]. This work focuses on developing a full coverage term discovery system.

Embedded segmental K-means (ES-KMeans) [18] model jointly optimizes both word boundaries and labels by altering between segmentation and clustering. ES-KMeans uses an acoustic word embedding method which uniformly downsamples the variable length segment to map it to a fixed dimensional vector. These fixed dimensional embeddings are used for clustering the speech segments. The idea is that a good embedding function would preserve the acoustic properties of the segments and acoustically similar segments would lie close together in fixed dimensional space.

ES-KMeans requires an initial subword boundary detection method that gives the location of probable word boundaries and removes unlikely word boundaries. The algorithm then clusters the initial segments and eliminates some of the initial boundaries based on the current word model (cluster centers). The final system performance depends heavily upon the quality of the initial boundaries. We propose to use unsupervised phoneme segmentation algorithm which produces boundaries that deviate less from the true boundaries as compared to the originally used syllable segmentation method [21]. The use of shorter acoustic felicitates finer refinements while searching for words. Both these reason contribute to higher system performance. A kernel Gram matrix based segmentation method is used for obtaining the initial phoneme boundaries in an unsupervised manner [7, 22]. A word would contain a larger number of phonemes than syllables, and this increases the number of possible combinations to check. To reduce the run time of the model, we use a non-linear dimensionality reduction method to map the finite dimensional input vector to a compact representation. It allows segments to be efficiently compared directly in the embedding space. The learned embeddings show substantial improvements in the run-time.

The effectiveness of the proposed approach is demonstrated on zero Resource speech challenge 2017. The proposed algorithm can scale to large datasets of size 45 hours. We conduct two sets of experiments. First, we analyze the impact of using the phoneme boundaries as pre-segmentation for ES-KMeans as opposed to initially used syllable boundaries. We also perform additional experiment to quantify the impact of initial segmentation on the final system performance. Second, we examine how the dimensionality of the autoencoder embedding affects the speed and accuracy of the ES-KMeans relative to MFCC based embeddings. The learned embeddings give similar performance to MFCC features while giving

10-15 times speed up in the runtime.

2. EMBEDDED SEGMENTAL K-MEANS

Given an utterance represented by the sequence of feature vectors $\mathbf{X} = (\mathbf{x_1}, \mathbf{x_2}, ..., \mathbf{x_N})$, Where $\mathbf{x_i}$ is the d dimensional feature vector, and N is the total number of frames, we aim to divide the utterance into word like segments and cluster these segments into a finite number of word types. ES-KMeans estimates both the segmentation and the cluster assignments. The algorithm uses a two-step iterative optimization procedure that successively optimizes word boundaries and the clustering.

The ES-KMeans algorithm begins with an initial segmentation method that divides the data into various variable length segments. An acoustic word embedding method [23, 24, 25] f_e maps the variable length segments to fixed dimensional vector i.e. segment $\mathbf{x_{t_1:t_2}}$ is mapped to a vector $y_i = f_e(x_{t_1:t_2})$. Using the segment boundaries, all the segments in the data are embedded and are represented by a set of vectors $\mathcal{Y} = \{y_i\}_{i=1}^{M}$. The segmentations for the data set are represented by $\mathcal{Q} = \{q_i\}_i^S$, where S is the number of utterances and q_i indicates the boundaries for utterance i. $\mathcal{Y}(\mathcal{Q})$ denotes the embeddings under the current segmentation. Now an objective function that depends on both segmentation and clustering assignments in required. Standard K-means [26] objective can extended to include both the segmentation Q and the cluster assignments z i.e. $\underset{Q,z}{min} \sum_{c=1}^{K} \sum_{y \in \mathcal{Y}_c \cap \mathcal{Y}(Q)} \|y - \mu_c\|^2$, where $\mathcal{Y}_c \cap \mathcal{Y}(\mathcal{Q})$ are the segments belonging to cluster c under segmentation $\mathcal{Q}$. But this objective function would favor longer segments, i.e., it will try to put entire utterance into a single segment to minimize the number of terms in the summation. Any boundary insertion would increase the number of terms in the summation and worsen the objective function. So we penalize large segments by weighting them by their duration. The objective function now becomes

$$\underset{Q,z}{min} \sum_{c=1}^{K} \sum_{x \in \mathcal{Y}_c \cap \mathcal{Y}(Q)} len(y) \|y - \mu_c\|^2 \tag{1}$$

Where len(y) is the number of frames in the segment which is represented by the embedding y. There is an additional parameter which controls the minimum length of the segment. The algorithm then freezes the segments boundaries to obtain the cluster assignments that minimize the objective function. Given the fixed dimensional representations of the segments, any standard clustering algorithm (e.g., K-Means) can be used to obtain the cluster assignments z. A dynamic programming algorithm then updates the word boundaries Q based on the current cluster centers and assignments z. The algorithm moves back and forth between optimizing the segmentation Q and cluster assignments z until some convergence criteria is reached.

3. INITIAL PHONETIC SEGMENTATION

The main idea behind the initial segmentation algorithm is that the frames from the same segment show higher degrees of similarity than those from different segments. A Gaussian kernel [27] is used for computing similarity between every pair of feature vectors. The kernel Gram matrix is computed as

$$G(i,j) = \exp\left(-\frac{||\mathbf{x_i} - \mathbf{x_j}||^2}{h}\right), \quad 1 \leq i,j \leq N \tag{2}$$

where $\mathbf{x_i}$, $\mathbf{x_j}$ are two feature vectors, $||.||$ denotes the Euclidean norm of a vector and h controls the width of the Gaussian kernel. 39-dimensional Mel-frequency cepstral coefficients are used for computing the Gram matrix G. The feature vectors taken from the same segment give rise to the block diagonal structure of the kernel Gram matrix. The speech segmentation task is viewed as identifying the square patches in the Gram matrix.

To identify the segment boundaries from the Gram matrix, a temporal neighbourhood criterion is used. Let ϵ-neighbourhood for the i^{th} frame $\mathbf{x_i}$ be the set of all the frames in the utterance whose distance to $\mathbf{x_i}$ is less than a predefined threshold ϵ. As the frame from the same segment as that of $\mathbf{x_i}$ would be acoustically similar to $\mathbf{x_i}$; they should also belong to the ϵ-neighbourhood for the i^{th} frame. All the immediate frames after the $\mathbf{x_i}$ that fall in ϵ-neighbourhood for the i^{th} frame are referred to as temporally reachable frames from $\mathbf{x_i}$. A segment can only contain consecutive frames so the boundary of the segment containing $\mathbf{x_i}$ frame can be located by finding the first temporally unreachable frame from $\mathbf{x_i}$. Relying on just one frame for boundary detection might lead to spurious boundaries. So, we check for τ consecutive points being unreachable from $\mathbf{x_i}$ for detecting the segment boundaries. However, a larger value of τ would result in missed detection, and a smaller value would result in false alarms. We use a value of $\tau = 3$ for boundary detection. The minimum and maximum possible acoustic segment lengths are restricted to 20 ms and 500 ms, respectively.

All the frames predict their respective endpoints and all the frames from the same segment would predict the same or nearby frames as endpoints. For each frame, we keep track of the number of frames that predicted it as their endpoint. The frames with a higher count than their adjacent frames are the final endpoints. The choice of ϵ affects the segmentation performance. Different segments exhibit varying degrees of similarity, e.g., voiced segments are generally more similar than the unvoiced segments. An adaptive ϵ that adjusts automatically according to the acoustic properties of the segment is required. The ϵ is set to be the running mean of the segment and once a boundary is detected the ϵ is reset.

4. EXPERIMENTAL EVALUATION

We use Zero Resource speech 2017 challenge for evaluating the performance of the proposed approach. The challenge aims to measure the robustness of the unsupervised term discovery systems across speakers and languages. The 2017 challenge dataset consists of 5 languages and contains more than 100 hours of data. The vast amount of data ensures that the term discovery systems are scalable to large speech corpora. Three languages English, French, and Mandarin are released along with the term discovery evaluation system for each of them. The system hyper-parameters should be optimized such that the systems generalize well across languages.

The evaluation kit uses various well-established metrics to quantify the system performance [28]. All the metrics assumes the availability of a time-aligned transcription of the speech data. Normalized edit distance (NED) measures the differences in the phoneme sequences of a word class, while the coverage (Cov) measures the fraction of the data covered by the discovered word like units. The token recall is the probability that a gold word (manual word transcription) token is found in obtained word classes. Token precision is the probability that a discovered word token would match a gold word token. A similar definition is used for calculation of type performance. The metric 'type' measures the correspondence between the discovered word and the true words in the data. The segmentation measures the quality of the boundaries of the identified word-

Embeedings	NLP		type			token			boundary			speed-up
	NED	Cov	P	R	F	P	R	F	P	R	F	
MFCC	88.4	117.2	11.2	13.1	12.1	11.3	9.9	10.6	58.4	52.8	55.5	1
Dim 20	87.2	117.2	11.5	13.5	12.4	11.5	10.0	10.7	58.9	52.6	55.6	10.3
Dim 15	96.4	117.2	10.4	10.2	10.3	10.6	7.2	8.6	64.3	47.9	54.9	13.3
Dim 10	89.4	117.2	11.0	12.2	11.5	11.3	9.2	10.1	60.4	51.4	55.6	15.4

Table 1: Effect of autoencoder bottleneck dimensionality on the final performance and run-time on Mandarin

like units with the manual word boundaries.

4.1. Compact acoustic embeddings

ES-KMeans relies on a simple downsampling based method for getting the fixed dimensional embedding. Uniform downsampling is achieved by dividing the segment into a fixed number of sub-segments. The average vectors of all these sub-segments are concatenated to obtain the final embedding. Uniform downsampling gives finite dimensional embeddings, but they are typically high dimensional, e.g., downsampling a segment into ten vectors with 39-dimensional MFCC as the input would result in a 390-dimensional feature vector. The use of phoneme for initializing ES-KMeans increases the number of potential word endpoints that the algorithm has to consider. Both these, high dimensional input and large number potential endpoints, reasons contribute to a significant increase in the computational cost of the algorithm. As ES-KMeans is agnostic to the input features, so we focus on learning a compact representation that can reduce the runtime without affecting the system performance. Here, we use a stacked autoencoder to project the finite dimensional embedding obtained using downsampling into a much smaller dimension. The autoencoder networks are data-driven and have been used for representation learning [29]. The bottleneck features extracted from autoencoders were shown to improve the performance of the speech systems[5, 30]. The autoencoder consists of two parts: an encoder which takes the downsampled segment as input and encodes it into a much smaller dimension, typically 10-20 and a decoder which tries to reconstruct the input from the encoded representation. The autoencoder tries to minimize the difference between the input and the reconstructed version of the input. The size the encoding dimension is a crucial parameter, it affects both the runtime and the final system performance. The smaller the encoding dimension, the lesser the run time of the algorithm but with the decrease in the size encoding dimension the reconstruction loss increases. It decreases the quality of the representations which lowers the final system performance. The segments are uniformly downsampled to obtain finite-dimensional vectors. We use these vectors as input while training the autoencoder. We use 80% of the data as training set and the remaining 20% as the validation set. Table 1 summarizes the term discovery performance with varying bottleneck dimension. For all the languages, the embedding dimension is fixed to 20. We can extract embeddings for originally used subwords, syllables, as well and reduce the run-time of the algorithm. These compact embeddings make the ES-KMeans scaleable to huge speech datasets and reduce the processing time of the algorithm.

4.2. Comparison of syllable and Phoneme initialization

The ES-Kmeans selects the optimal boundary from the set of initial boundaries. The discovered boundaries should be as close as possible, ideally coincide, to the true word boundaries. Figure 1, compares the boundaries discovered by the syllable and the phoneme segmentation algorithm. The phonetic boundaries are closer to true word boundaries. The quality of the acoustic embeddings depends on the segmentation. A poor segmentation would produce lower quality embeddings which dampen the performance of the clustering step. The subsequent segmentation update depends on the clustering. So the quality of the finally discovered words would depend on the initial segmentation. To measure the impact of the initial segmentation on the final performance, we conducted the following experiment. We measured the initial segmentation performance w.r.t true boundaries and the final type and token F-scores obtained by the system. As evident from Table 2, for all three languages there is a direct correlation between the initial word boundary performance and the final type/token accuracy. For Mandarin, the phoneme segmentation achieves a higher F-score 43.9 than the syllable-based segmentation 39.9 which is reflected in higher type/token F-score for the phoneme-based method. For English, on the other hand, the syllable segmentation has better initial segmentation performance which leads to higher type/token F-score for the syllable-based method.

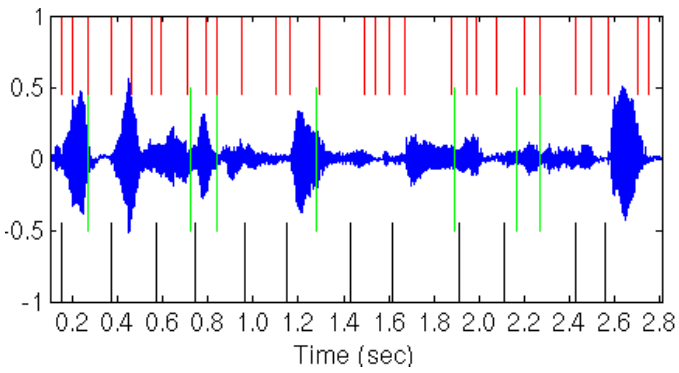

Fig. 1: Comparison of the initial boundaries obtained using various algorithms on Mandarin dataset. The phonetic boundaries, the syllable boundaries and the true word boundaries are shown Red, Black and Green respectively. The figure is best viewed in color.

Language	P	R	F	type (F)	Token (F)
Mandarin (phn)	29.3	87.3	43.9	8.8	8.7
syl	34.3	47.7	39.9	3.1	2.9
French (phn)	21.6	69.8	33.0	5.5	5.9
syl	20.5	29.8	24.3	4.2	3.7
English (phn)	22.8	71.4	34.6	6.1	6.2
syl	33.0	46.4	38.6	11.1	13.5

Table 2: The initial segmentation performance and the quality of the finally discovered words

Regardless of the accuracy of the discovered boundaries both the methods produce some boundaries that deviate from the true bound-

Language	System	NLP		type			token			boundary		
		NED	Cov	P	R	F	P	R	F	P	R	F
English (45 hours)	Baseline	**30.7**	2.9	4.5	0.1	0.2	4.0	0.1	0.1	37.5	0.9	1.8
	ES-KMeans	72.6	100	**8.3**	**16.7**	**11.1**	**13.0**	**14.1**	**13.5**	**51.0**	**54.4**	**52.7**
	Proposed	72.2	**100.9**	4.5	9.4	6.1	5.0	8.2	6.2	26.4	41.2	32.2
French (24 hours)	Baseline	**25.4**	1.6	**6.9**	0.2	0.3	**5.2**	0.1	0.1	30.9	0.6	1.1
	ES-KMeans	67.3	97.2	3.1	6.3	4.2	3.5	3.9	3.7	**37.8**	**41.6**	**39.6**
	Proposed	68.1	**97.5**	4.2	**7.9**	**5.5**	4.8	**7.6**	**5.9**	25.4	38.4	30.6
Mandarin (2.5 hours)	Baseline	**30.7**	2.9	4.5	0.1	0.2	4.0	0.1	0.1	37.5	0.9	1.8
	ES-KMeans	88.1	100	2.5	4.1	3.1	2.5	3.4	2.9	36.5	47.1	41.1
	Proposed	80.0	**117.5**	**7.7**	**10.4**	**8.8**	**6.9**	**11.5**	**8.7**	**43.8**	**66.8**	**52.9**

Table 3: Performance of the baseline system, syllable based ES-KMeans and the proposed phoneme based ES-KMeans on the three languages of Zero Resource Speech Challenge 2017

aries. The shorter units, phonemes, allow finer adjustments while discovering words and find words that are closet to the true words. We measure the deviation between the boundaries of the finally discovered words from both segmentation methods and the boundaries of the nearest true words. The word discovered using phoneme segmentation diverge much less from the true boundaries as compared to syllable based method, Figure 2. More than half the discovered words using phoneme segmentation have boundaries within 10ms of true word.

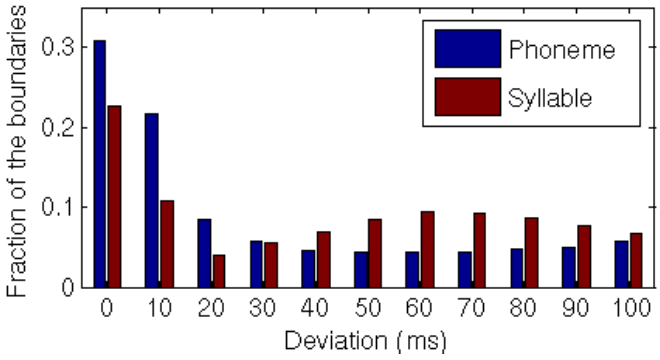

Fig. 2: The deviation of the final word boundaries, obtained using phoneme and syllable initial segmentation, from the true word boundaries on Mandarin dataset.

4.3. Zero Resource 2017: Full system performance

For each of the languages, the same set of hyper-parameters are used for term discovery. System performance varies significantly across languages because the exact same system is used across languages. The optimal set of parameters might be different across languages. As evident from the table 3, our algorithm performs well across languages. The baseline system [16] finds high precision isolated segments. This high precision is achieved by discarded a lot of the discovered segments as background noise. This results in very low coverage. The baseline system performs better only terms of NED which is computed only on the discovered patterns. We, on the other hand, cover the whole data which results in the higher word boundary, word token, and type performance. The words discovered by our algorithm are closer to true words (type F-score) for all the languages as compared to the baseline approach. The choice of initial segmentation plays an important role on the system performance (see section 4.2 for details) . The syllable initialized ES-KMeans performs better on English, whereas the phoneme initialized method performs very well on Mandarin. For French, the phoneme-based method achieves better word token, type F-scores. We conducted a set of experiments with different parameters (e.g varying minimum word length) across languages. We observed that the best performance parameters differ for languages. The system that worked well for all the languages results are used for comparison with other existing methods.

5. CONCLUSIONS AND FUTURE WORK

This paper proposes a new unsupervised phoneme based initialization method for ES-KMeans. The phoneme boundaries are closer to true boundaries which leads to increment in the system performance. We propose to use a dimensionality reduction method to tackle the computational needs associated with the use of shorter sub-word units like phonemes. The proposed method significantly outperforms the baseline method. The dimensionality reduction method can be used with any input features. The learned compact embeddings are 10-15 times faster than the MFCC embeddings while giving the same performance. Faster methods would allow us to do detailed experiment with parameters and come up with the optimal set of parameters. We also established a correlation between the initial segmentation performance and the final term discovery performance. In future, we would focus on improving the quality of initial subword boundaries. The difference in performance across languages (for both syllable and phoneme-based methods) might be due to the fact that different languages have different word length distribution and we are using a fixed minimum word length across languages. In future, our focus would be on an automatic estimation of system parameters for a language in an unsupervised manner. This would allow us to develop the best system for a language instead of a system that works reasonably well across all the languages.

6. ACKNOWLEDGEMENT

First and third authors would like to acknowledge Ministry of Human Resource Development (MHRD), Government of India, for sponsoring this work under IMPRINT initiative.

7. REFERENCES

[1] M. Versteegh, R. Thiolliere, T. Schatz, X.-N. Cao, X. Anguera, A. Jansen, and E. Dupoux, "The zero resource speech challenge 2015.," in *INTERSPEECH*, pp. 3169–3173, 2015.

[2] M. Versteegh, X. Anguera, A. Jansen, and E. Dupoux, "The zero resource speech challenge 2015: Proposed approaches and results," *Procedia Computer Science*, vol. 81, pp. 67–72, 2016.

[3] P. K. Kuhl, "Early language acquisition: cracking the speech code," *Nature reviews neuroscience*, vol. 5, no. 11, pp. 831–843, 2004.

[4] Y. Zhang, R. Salakhutdinov, H.-A. Chang, and J. Glass, "Resource configurable spoken query detection using deep boltzmann machines," in *Acoustics, Speech and Signal Processing (ICASSP), 2012 IEEE International Conference on*, pp. 5161–5164, IEEE, 2012.

[5] L. Badino, C. Canevari, L. Fadiga, and G. Metta, "An auto-encoder based approach to unsupervised learning of subword units," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pp. 7634–7638, IEEE, 2014.

[6] N. Zeghidour, G. Synnaeve, N. Usunier, and E. Dupoux, "Joint learning of speaker and phonetic similarities with siamese networks.," in *INTERSPEECH*, pp. 1295–1299, 2016.

[7] S. Bhati, S. Nayak, and K. S. R. Murty, "Unsupervised speech signal to symbol transformation for zero resource speech applications," *Proc. Interspeech 2017*, pp. 2133–2137, 2017.

[8] H. Gish, M.-h. Siu, A. Chan, and B. Belfield, "Unsupervised training of an hmm-based speech recognizer for topic classification," in *Tenth Annual Conference of the International Speech Communication Association*, 2009.

[9] M. Huijbregts, M. McLaren, and D. Van Leeuwen, "Unsupervised acoustic sub-word unit detection for query-by-example spoken term detection," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*, pp. 4436–4439, IEEE, 2011.

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

[11] B. Varadarajan, S. Khudanpur, and E. Dupoux, "Unsupervised learning of acoustic sub-word units," in *Proceedings of the 46th Annual Meeting of the Association for Computational Linguistics on Human Language Technologies: Short Papers*, pp. 165–168, Association for Computational Linguistics, 2008.

[12] M.-h. Siu, H. Gish, A. Chan, W. Belfield, and S. Lowe, "Unsupervised training of an hmm-based self-organizing unit recognizer with applications to topic classification and keyword discovery," *Computer Speech & Language*, vol. 28, no. 1, pp. 210–223, 2014.

[13] S. Bhati, "Unsupervised spoken term discovery for zero resource speech processing," Master's thesis, Indian Insitute of Technology Hyderabad, 2017.

[14] H. Li, B. Ma, and C.-H. Lee, "A vector space modeling approach to spoken language identification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 15, no. 1, pp. 271–284, 2007.

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

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

[17] M. Sun *et al.*, "Joint training of non-negative tucker decomposition and discrete density hidden markov models," *Computer Speech & Language*, vol. 27, no. 4, pp. 969–988, 2013.

[18] H. Kamper, K. Livescu, and S. Goldwater, "An embedded segmental k-means model for unsupervised segmentation and clustering of speech," *arXiv preprint arXiv:1703.08135*, 2017.

[19] C.-y. Lee, T. J. O'Donnell, and J. Glass, "Unsupervised lexicon discovery from acoustic input," *Transactions of the Association for Computational Linguistics*, vol. 3, pp. 389–403, 2015.

[20] H. Kamper, A. Jansen, and S. Goldwater, "Unsupervised word segmentation and lexicon discovery using acoustic word embeddings," *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)*, vol. 24, no. 4, pp. 669–679, 2016.

[21] O. Räsänen, G. Doyle, and M. C. Frank, "Unsupervised word discovery from speech using automatic segmentation into syllable-like units.," in *INTERSPEECH*, pp. 3204–3208, 2015.

[22] S. Bhati, S. Nayak, and K. S. R. Murty, "Unsupervised segmentation of speech signals using kernel-gram matrices," in *National Conference on Computer Vision, Pattern Recognition, Image Processing and Graphics*, Springer, 2017.

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

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

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

[26] J. A. Hartigan and M. A. Wong, "Algorithm as 136: A k-means clustering algorithm," *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, vol. 28, no. 1, pp. 100–108, 1979.

[27] J.-P. Vert, K. Tsuda, and B. Schölkopf, "A primer on kernel methods," *Kernel Methods in Computational Biology*, pp. 35–70, 2004.

[28] B. Ludusan, M. Versteegh, A. Jansen, G. Gravier, X.-N. Cao, M. Johnson, and E. Dupoux, "Bridging the gap between speech technology and natural language processing: an evaluation toolbox for term discovery systems," in *Language Resources and Evaluation Conference*, 2014.

[29] Y. Bengio, A. Courville, and P. Vincent, "Representation learning: A review and new perspectives," *IEEE transactions on pattern analysis and machine intelligence*, vol. 35, no. 8, pp. 1798–1828, 2013.

[30] D. Renshaw, H. Kamper, A. Jansen, and S. Goldwater, "A comparison of neural network methods for unsupervised representation learning on the zero resource speech challenge.," in *INTERSPEECH*, pp. 3199–3203, 2015.