Unsupervised Segmentation of Speech Signals Using Kernel-Gram Matrices

***Abstract*—The objective of this paper is to develop an unsupervised method for segmentation of speech signals into phoneme-like units. The proposed algorithm is based on the observation that the feature vectors from the same segment exhibit higher degree of similarity than the feature vectors across the segments. The kernel-Gram matrix of an utterance is formed by computing the similarity between every pair of feature vectors in the Gaussian kernel space. The kernel-Gram matrix consists of square patches, along with the principle diagonal, corresponding to different phoneme-like segments in the speech signal. It detects the number of segments, as well as their boundaries automatically. The proposed approach does not assume any information about input utterances like exact distribution of segment length or correct number of segments in an utterance. The proposed method out-performs the state-of-the-art blind segmentation algorithms on Zero Resource 2015 databases and TIMIT database.**

***Index Terms*—Blind segmentation, Gaussian kernel, Kernel-Gram matrix, Phonetic segmentation**

I. Introduction

Speech signal can be considered as a sequence of basic phonemic units. Most of the automatic speech recognition (ASR) systems depend on accurate recognition of these basic units. Even, the speech synthesis systems rely on synthesizing acoustic waveforms for these basic units. Automatic segmentation of speech into phoneme-like units plays an important role in several speech applications including speech recognition, speech synthesis and audio search [1]–[3]. Segmentation of speech forms a crucial initial step in the acoustic segment modelling which finds applications in audio search [4] and unsupervised transfer learning [5]. Speech segmentation in an unsupervised manner is an important step in zero-resource speech processing [6]. Segmentation of speech into basic units involves detecting the time instants at which the vocal tract system transitioned from one state to another state. However, the transition is not abrupt, rather it happens in a continuum, making it difficult to unambiguously detect the time instant of transition. As a result, accurate identification of phoneme boundaries is difficult even for human beings. There could be differences between the boundary locations marked by two different experts.

Automatic approaches to speech segmentation can be broadly categorized into supervised and unsupervised methods. Supervised methods are mainly model-based, and employ probabilistic models of the phonemes to classify the acoustic regions into phonemes. Hidden Markov models (HMM) are typically used to model the acoustic features extracted from the phonemes [7]. If sequence of phones is available for an utterance then the boundaries can be determined with forced alignment using Viterbi algorithm [8]. Though the supervised approaches deliver high performance, they require a large amount of manually transcribed data. Moreover, phoneme models are specific to the language on which they are trained, and cannot be readily used to segment speech data from other languages. Segmentation methods that fine-tune parameters on a validation set suffer from the same problem. These approaches can not be applied to a new target language without the required amount of manual supervision.

Unsupervised approaches to speech segmentation, on the other hand, do not require manually transcribed speech data. In fact, it has been argued that unsupervised segmentation is similar to the perception of speech by infants [9]. Unsupervised approaches are metric-based, and use the distance between the adjacent regions to detect the change points in the audio stream [10]. Support vector machines were employed to detect abrupt spectral changes which were marked as segment boundaries [11]. Maximum spectral transition was used for detecting boundaries [12] and a "jump function" was proposed for identifying changes in audio signal [13]. Segmentation has also been formulated as a clustering problem [14]–[16]. Boundary models were used for capturing the characteristics of the signal around a boundary [17], [18]. Segmentation by recognition was proposed where a probabilistic model was used to determine whether a feature belongs to a segment or not [19]. An agglomerative algorithm was proposed for speech segmentation, which begins with as many numbers of segments as the number of frames in the speech signal, and successively combines the pair of most similar adjacent segments in each iteration, till the required number of segments is met [20]. Time-frequency speech spectrograms were used for speech segmentation and the intensity changes in spectrogram were hypothesized as potential change points [21]. Spectrogram and Mel cepstral coefficients were used for segmentation and the results were combined from both the features to give final segment end points [22]. Microcanonical Multiscale Formalism (MMF) was used for segmentation which analyzes the local dynamics of speech from a multiscale perspective [23]. A 2-D filter that moves along diagonal in correlation matrix obtained using FFT features was proposed for estimating the segment boundaries [24]. A non-parametric Bayesian approach was proposed for acoustic model discovery [25].

In this paper, a local similarity measure for unsupervised phonetic segmentation of speech signals is proposed which is motivated by the observation of distance matrix, shown in Fig. 1. This depicts the distance between every pair of feature vectors extracted from the speech signal. Features from the

same segment exhibit higher degree of similarity (or lesser distance), than the features from two different segments. From the distance matrix, the sequence of features having a higher degree of similarity are separated and labeled as a segment. Since we process the features extracted from the signal in a sequential manner, it does not require the entire signal at the same time. Moreover, the number of segments can be automatically determined from speech signal.

The rest of the paper is organized as follows: Section II discusses the proposed similarity measure for unsupervised phonetic segmentation of speech. Section III presents experimental evaluation and effectiveness of the proposed method. Finally, paper is concluded in Section IV.

II. Segment detection from kernel-Gram matrix

Let the sequence of states of the vocal-tract system during the production of a speech signal be represented by a sequence of feature vectors $\mathbf{X} = (\mathrm{x}_1, \mathrm{x}_2, \ldots, \mathrm{x}_N)$, where x_i denotes the d-dimensional feature vector extracted from i^{th} frame of the speech signal, and N is the total number of frames. The objective of speech segmentation is to divide the sequence $\mathbf{X}$ into K non-overlapping contiguous segments $\mathbf{S} = (\mathrm{s}_1, \mathrm{s}_2, \ldots \mathrm{s}_K)$, where s_j denotes j^{th} segment that begins at frame b_j and ends at frame e_j. The segmentation algorithm should ensure that feature vectors in each of the segment s_j are acoustically similar, and represent a phoneme-like unit. Hence the segmentation algorithm should determine the number of segments and detect their beginning and end points from the acoustic similarity of the feature vectors $\mathbf{X}$.

In the absence of any information about the source distribution, we propose to detect the segment boundaries from the Gram matrix obtained using Gaussian kernel [26]. We assume that two feature vectors from the same segment must have a higher degree of similarity than two feature vectors from different segments. This assumption is justified as the feature vectors from the same segment are drawn from the same source distribution, while the feature vectors from different segments are drawn from different source distributions.

In the proposed approach, the similarity between two feature vectors x_i and x_j is computed using Gaussian kernel as

$$G(i,j) = \exp\left(-\frac{||\mathrm{x}_i - \mathrm{x}_j||}{h}\right) \tag{1}$$

where $||.||$ denotes the Euclidean norm of a vector and h is a free parameter which can be used to adjust the width of the Gaussian kernel. Kernel-Gram matrix G can be obtained by computing the similarity between every pair of feature vectors in the sequence $\mathbf{X}$.

Gram matrix computed from 13-dimensional Mel-frequency cepstral coefficient (MFCC) features, extracted from a speech utterance is shown in Fig. 1(a). The intensity of a pixel at (i,j) indicates the similarity $G(i,j)$ between the feature vectors x_i and x_j. The region around the principal diagonal corresponds to temporally closer segments. The square patches of higher degree of similarity along the diagonal correspond to acoustically similar segment. Manually marked phoneme boundaries are also shown in the Fig. 1. It is observed that the manually marked boundaries, shown in red colour, coincide exactly with the square patches along the principal diagonal. In this work, the task of speech segmentation is equivalent to identifying the square patches along the main diagonal of the Gram matrix.

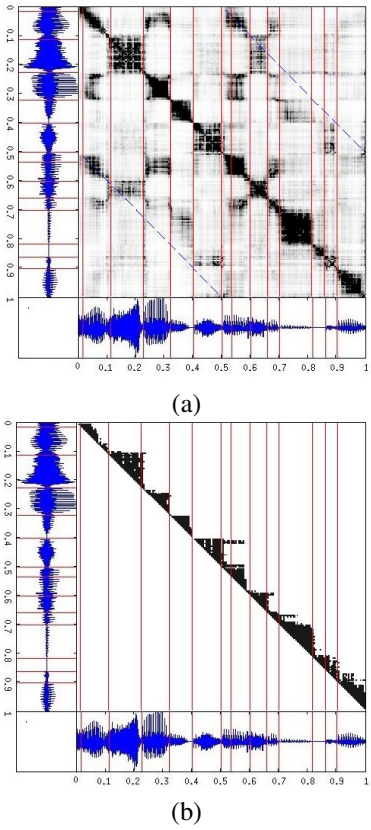

Fig. 1. Illustration of Similarity matrices. (a) Kernel-Gram matrix between every pair of feature vectors extracted from the speech signal. (b) End point detected by each frame. Red lines indicate manually marked phoneme boundaries.

As segment boundaries occur in a small neighborhood around the diagonal, the search space can be restricted to a small region parallel to the diagonal. This is analogous to constraining the dynamic time warping path using Itakura parallelogram [27]. The length constraints are shown in Fig. 1(a) with dotted lines. As Gram matrix is symmetric, it is enough to compute the upper triangular portion of the Gram matrix. These two constraints lead to significant reduction in computational complexity. The similarity values are higher when the column index j is close to the row index i, i.e., around diagonal, indicating that the frames i and j belong to same segment. On the other hand, as the column index j moves away from the row index i, the similarity values decrease indicating that the frames i and j belong to different segments.

A density-based algorithm is used for identifying the segment boundaries from the kernel gram matrix. We share some concepts such as reachability and ϵ neighbourhood with DBSCAN [28] algorithm which are used in the calculation of

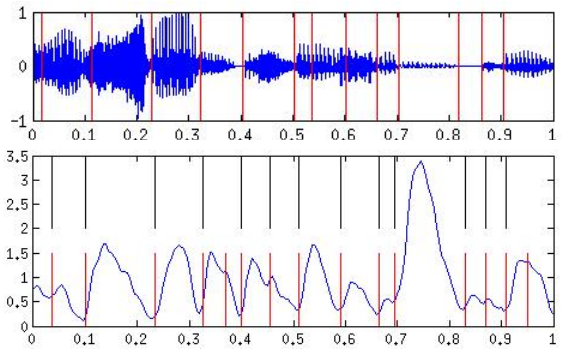

Fig. 2. Top - Speech signal with manually marked boundaries. Bottom - Segment profile with detected boundaries shown in Red lines. Black lines show manually marked boundaries.

segment boundaries.

A frame, x_j, is in ϵ neighbourhood of x_i if $1 - G(i,j) < \epsilon$. Given an appropriate ϵ, all the features that are from the same segment as that of x_i will be in ϵ neighbourhood of x_i. Since segments are continuous only consecutive frames can be in a segment. For frame x_i, we check consecutive frames for the neighbourhood and maintain a run length (number of neighbours) l_i for each frame.

A frame x_{i+l_i} is temporally reachable from x_i, if all the frames in between x_i, x_{i+l_i} are in ϵ neighbourhood of frame x_i. The first temporally unreachable frame from x_i can be considered as the end of the segment containing the x_i frame. However, making a decision with just one single frame could be erroneous due to noise in data. Hence, we use K-step temporal unreachablity, i.e., K successive frames being unreachable, to identify the boundary of the segment containing x_i. The chances of making an erroneous estimate decrease with increase in K. So for a frame x_i, the end point is estimated at $i + l_i$ if and only if x_{i+l_i} is temporally reachable by x_i and all the frames $x_{i+l_i+1}, \ldots, x_{i+l_i+K}$ are not in ϵ neighbourhood of x_i.

A frame in the beginning of the segment will have the highest number of neighbours and a frame, in the end, will have the lowest number of neighbours. For each frame, we construct the neighbourhood graph, which is basically the sum of all the similarities in the K-point reachablity of that frame. Neighbourhood graph is given as

$$N_G(i) = \Sigma_{j=i}^{l_i} G(i,j) \tag{2}$$

where l_i is the run length of x_i frame. For a frame, in the beginning, the value of N_G will be highest since the maximum number of frames contribute to it. As we move towards the end of the segment the value of N_G will keep decreasing as shown in Fig. 2. At the end point, the N_G will be minimum, ideally zero. The location of minimas in N_G gives the location of end points of all the segments in the given utterance.

Given the reachability threshold ϵ and parameter K in K-step temporal unreachability, any utterance can be segmented into phoneme-like units. The segmentation performance critically depends on the choice of ϵ. Acoustic properties of the segments differ across segments. For example, a frame taken from voiced segment will be more similar to another frame taken from voiced segment as compared to a frame taken from an unvoiced segment. This makes finding a global ϵ very hard which is consistent across different segments. So, we use an ϵ value that adapts itself according to the acoustic properties of the segment.

To automatically determine the value of ϵ and to allow different segments to use separate ϵ threshold, we develop a simple algorithm. The algorithm is based on the observation that similarity between frames of the same segment is higher than the average similarity. New threshold for x_i frame is given as

$$\epsilon_i = \frac{\sum_{j=i}^{\tau} G(i,j)}{\tau} \tag{3}$$

where τ is the diagonal window constraint. For each frame, a different ϵ is computed automatically using the acoustic properties of the segment in consideration.

After getting segment boundaries, minimum length criteria is used for avoiding segments that are not possible. Minimum length is fixed to be 20 ms. K is chosen to be equal to the minimum length of the segment because if there is a segment boundary then at least minimum length number of frames will be temporally unreachable after the boundary. For longer segments, the number of unreachable points will exceed minimum length.

III. Experimental Evaluation

The speech segmentation algorithm proposed in the paper was evaluated on TIMIT [29] and Zero Resource 2015 databases: Tsonga and English [30]. The following sections explain the performance evaluation on both the benchmarks. The kernel width is simply kept 1 and the algorithm determines the value of ϵ automatically.

A. Experimental Evaluation on TIMIT

TIMIT dataset has been used for evaluation segmentation algorithms [12], [21]–[24]. All the sentences were manually transcribed and segmented at phone level using 61 phone labels. In this work, we have used MFCC features to represent the state of the vocal-tract system at a given instant of time. The segment boundaries are extracted from the kernel-Gram matrix, computed from the MFCC features, as discussed in Section II. Let N_C be the number of correctly detected boundaries (within a given tolerance interval), N_T is the total number of detected boundaries and N_G is the total number manual boundaries. The performance of the proposed algorithm is evaluated, by comparing the detected boundaries with the manually marked boundaries, using the following intermediate metrics:

- Hit Rate (HR) : It is the fraction of reference boundaries that are correctly detected (N_C/N_G). It is also called recall rate of the segmentation system.
- Over Segmentation (OS) : It represents how many extra (less) boundaries are detected as compared to reference boundaries ($(N_T - N_G)/N_G$).

TABLE I
PERFORMANCE COMPARISON OF SPEECH SEGMENTATION ALGORITHMS FOR 20 MS TOLERANCE WINDOW. THE ∗ MARK REPRESENTS USE OF A VALIDATION SET FOR PARAMETER FINE TUNING.

method	F	R
Kernel Width (h = 1)	0.76	0.79
Dusan et. al. [12]	0.71	0.73
Khanagha et. al. [23]	0.74	0.77
Adriana et. al. [22] *	0.76	0.80
Leow et. al. [21] *	0.75	0.78
Rasanen et. al. [24] *	0.76	0.78

TABLE II
RESULTS (IN PERCENTAGE) FOR STD TASK ON ZEROSPEECH 2015 DATABASES: ENGLISH AND XITSONGA (IN BRACKETS). THE BEST SCORES FOR EACH EVALUATION METRIC ARE HIGHLIGHTED IN BOLD.

System	boundary		
	Precision	Recall	F-score
Baseline [31]	44.1 (22.3)	4.7 (5.6)	8.6 (8.9)
Vseg [32]	**76.1** (26.2)	28.5 (26.3)	41.4 (26.3)
EnvMin [32]	75.7 (16.3)	27.4 (24.4)	40.3 (19.5)
Osc [32]	75.7 (**29.2**)	33.7 (39.4)	46.7 (33.5)
CC-PLP [33]	39.6 (19.4)	7.5 (11.2)	12.7 (14.2)
CC-FDPLS [33]	35.4 (18.8)	38.5 (64)	36.9 (29)
Proposed	41.2 (22.5)	**71.1** (**74.8**)	**52.2** (**34.6**)

- False Alarm (FA) : The fraction of incorrectly detected boundaries ($(N_T - N_C)/N_T$).

The overall quantification of segmentation algorithm is done with a global measure, F score, which combines all the intermediate scores.

$$F = \frac{2 * (1 - FA) * HR}{1 - FA + HR} \tag{4}$$

There is another global measure, R, which emphasizes more on over segmentation (OS). It argues that recall rate can be increased by inserting random boundaries without changing the algorithm.

$$r_1 = \sqrt{(1 - HR)^2 + (OS)^2};\ r_2 = \frac{-OS + HR - 1}{\sqrt{2}} \tag{5}$$

The final metric is defined as

$$R = 1 - \frac{|r_1| + |r_2|}{2} \tag{6}$$

We use metrics R and F for evaluating the segmentation algorithm. The performance of the proposed algorithm is given in Table I. For $h = 1$, approximately 73% of the detected boundaries fall within the 20 ms tolerance interval from the manually marked boundaries.

The agglomerative algorithm proposed by Qiao et. al, requires the number of expected segments as input [20]. This method uses manual transcriptions for calculating the exact number of segments for the input utterance. The neural network based segmentation method proposed by Vuuren et. al. [34] used transcriptions for entire train data to learn the probability distribution of segment lengths. Both these approaches achieve very high performance but due to their strong prior requirements, recent works [22], [23] have put them in the category of semi-supervised approaches and performance comparison is done only with zero or minimal fine tuning approaches. We follow the same practice. Adriana et. al. [22] used a small validation set for adjusting the minimum peak height in probability function. Also, the beginning and end silence regions were trimmed to 50 ms which contribute a high number of spectral discontinuities in input signal. Leow et. al. [21] found the best performing system by evaluating the performance and then choosing the parameters of the best system. In the proposed method, the kernel width is simply kept 1. The proposed algorithm selects the optimal number of segments automatically.

B. Zero Resource 2015

We also evaluate the performance of the proposed segmentation method on the zero speech challenge 2015 datasets. This dataset consists of 10.5 hours of casual conversations in American English, and 5 hours of read speech in Xitsonga. The aim of the challenge (Track 2) was to discover recurring speech patterns in an unsupervised manner. Evaluation kit for measuring quality of discovered sub-words were provided as part of the challenge. In the present work, we only evaluate the segmentation performance. Recall measures the probability of finding a manual boundary within 30 ms of a discovered boundary. Precision measures the probability that a discovered boundary is within 30 ms of a manual boundary. The F-score is the harmonic mean of precision and recall. If the algorithm predicts boundaries only where manual boundaries are, then both precision and recall will be 1. The recall can be increased by predicting more boundaries but that would decrease the precision. Similarly, precision can be increased by predicting limited number of boundaries. The precision and recall can be traded off for each other. The F-score combines both of them and is used as a global measure for segmentation evaluation.

IV. SUMMARY & CONCLUSION

This study presents a new method for segmenting speech signals into phoneme like segments and a quantitative analysis between automatically detected boundaries and manually segmented boundaries on TIMIT database and Zero Resource 2015 database. The proposed algorithm achieves very high performance regardless of the language, recording environments or length of the utterances. In the proposed approach, all the frames estimate the end point of the segment they belong to. The individual decisions are then combined to give final end point decision. It reduces the deviation between predicted boundaries and manual boundaries. This is analogous to ensemble learning where several weak classifiers are combined to achieve performance on par with a single strong classifier. The proposed algorithm achieves the best performance among the blind segmentation algorithm on both databases.

Factors such as K point unreachability, adaptive threshold for each frame and combination of several end points improve the overall performance of the algorithm. We believe that performance can be improved by using more information rich features. The method presented in the paper can also be used in other sequence segmentation problems.

REFERENCES

[1] L. R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[2] E. Moulines and F. Charpentier, "Pitch-synchronous waveform processing techniques for text-to-speech synthesis using diphones," *Speech communication*, vol. 9, no. 5-6, pp. 453–467, 1990.

[3] S. Furui, *Digital speech processing: synthesis, and recognition*. CRC Press, 2000.

[4] A. Wang *et al.*, "An industrial strength audio search algorithm." in *Ismir*, vol. 2003. Washington, DC, 2003, pp. 7–13.

[5] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on knowledge and data engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.

[6] M. J. Gales and S. J. Young, "Robust continuous speech recognition using parallel model combination," *IEEE Transactions on Speech and Audio Processing*, vol. 4, no. 5, pp. 352–359, 1996.

[7] F. Brugnara, D. Falavigna, and M. Omologo, "Automatic segmentation and labeling of speech based on hidden markov models," *Speech Communication*, vol. 12, no. 4, pp. 357–370, 1993.

[8] K. Demuynck and T. Laureys, "A comparison of different approaches to automatic speech segmentation," in *International Conference on Text, Speech and Dialogue*. Springer, 2002, pp. 277–284.

[9] O. Scharenborg, M. Ernestus, and V. Wan, "Segmentation of speech: Child's play?" 2007.

[10] D. Rybach, C. Gollan, R. Schluter, and H. Ney, "Audio segmentation for speech recognition using segment features," in *Acoustics, Speech and Signal Processing, 2009. ICASSP 2009. IEEE International Conference on*. IEEE, 2009, pp. 4197–4200.

[11] M. Davy and S. Godsill, "Detection of abrupt spectral changes using support vector machines an application to audio signal segmentation," in *Acoustics, Speech, and Signal Processing (ICASSP), 2002 IEEE International Conference on*, vol. 2. IEEE, 2002, pp. II–1313.

[12] S. Dusan and L. Rabiner, "On the relation between maximum spectral transition positions and phone boundaries," in *Ninth International Conference on Spoken Language Processing*, 2006.

[13] G. Aversano, A. Esposito, and M. Marinaro, "A new text-independent method for phoneme segmentation," in *Circuits and Systems, 2001. MWSCAS 2001. Proceedings of the 44th IEEE 2001 Midwest Symposium on*, vol. 2. IEEE, 2001, pp. 516–519.

[14] M. M. Goodwin and J. Laroche, "Audio segmentation by feature-space clustering using linear discriminant analysis and dynamic programming," in *Applications of Signal Processing to Audio and Acoustics, 2003 IEEE Workshop on.* IEEE, 2003, pp. 131–134.

[15] Y. P. Estevan, V. Wan, and O. Scharenborg, "Finding maximum margin segments in speech," in *Acoustics, Speech and Signal Processing, 2007. ICASSP 2007. IEEE International Conference on*, vol. 4. IEEE, 2007, pp. IV–937.

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

[17] P. Micallef and T. Chilton, "Automatic identification of phoneme boundaries using a mixed parameter model," in *Fifth European Conference on Speech Communication and Technology*, 1997.

[18] J. P. van Santen and R. Sproat, "High-accuracy automatic segmentation." in *EUROSPEECH*, 1999.

[19] J. W. Chang and J. R. Glass, "Segmentation and modeling in segment-based recognition," in *Fifth European Conference on Speech Communication and Technology*, 1997.

[20] Y. Qiao, N. Shimomura, and N. Minematsu, "Unsupervised optimal phoneme segmentation: Objectives, algorithm and comparisons," in *Acoustics, Speech and Signal Processing, 2008. ICASSP 2008. IEEE International Conference on*. IEEE, 2008, pp. 3989–3992.

[21] S. J. Leow, E. S. Chng, and C.-H. Lee, "Language-resource independent speech segmentation using cues from a spectrogram image," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 5813–5817.

[22] A. Stan, C. Valentini-Botinhao, B. Orza, and M. Giurgiu, "Blind speech segmentation using spectrogram image-based features and mel cepstral coefficients," in *Spoken Language Technology Workshop (SLT), 2016 IEEE*. IEEE, 2016, pp. 597–602.

[23] V. Khanagha, K. Daoudi, O. Pont, and H. Yahia, "Phonetic segmentation of speech signal using local singularity analysis," *Digital Signal Processing*, vol. 35, pp. 86–94, 2014.

[24] O. Rasanen, U. Laine, and T. Altosaar, "Blind segmentation of speech using non-linear filtering methods," in *Speech Technologies*. InTech, 2011.

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

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

[27] L. R. Rabiner, *Multirate digital signal processing*. Prentice Hall PTR, 1996.

[28] M. Ester, H.-P. Kriegel, J. Sander, X. Xu *et al.*, "A density-based algorithm for discovering clusters in large spatial databases with noise." in *Kdd*, vol. 96, no. 34, 1996, pp. 226–231.

[29] J. S. Garofolo, L. F. Lamel, W. M. Fisher, J. G. Fiscus, and D. S. Pallett, "Darpa timit acoustic-phonetic continous speech corpus cd-rom. nist speech disc 1-1.1," *NASA STI/Recon technical report n*, vol. 93, 1993.

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

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

[32] O. Räsänen, G. Doyle, and M. C. Frank, "Unsupervised word discovery from speech using automatic segmentation into syllable-like units," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[33] V. Lyzinski, G. Sell, and A. Jansen, "An evaluation of graph clustering methods for unsupervised term discovery," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[34] V. Vuuren, L. Bosch, and T. Niesler, "Unconstrained speech segmentation using deep neural networks," in *ICPRAM 2015proceedings of the international conference on pattern recognition applications and methods*, vol. 1, 2015, pp. 248–254.