A Density Peak Clustering Approach to Unsupervised Acoustic Subword Units Discovery

Jia Yu*, Lei Xie*, Xiong Xiao†, Eng Siong Chng†, Haizhou Li‡
* Shaanxi Provincial Key Laboratory of Speech and Image Information Processing,
School of Computer Science, Northwestern Polytechnical University, Xi'an, China
† School of Computer Engineering, Nanyang Technological University, Singapore
‡ Institute for Infocomm Research, A*STAR, Singapore
E-mail: {jiayu,lxie}@nwpu-aslp.org, {xiaoxiong,ASESChng}@ntu.edu.sg, hli@i2r.a-star.edu.sg

Abstract—**This paper studies unsupervised acoustic units discovery from unlabelled speech data. This task is usually approached by two steps, i.e., partitioning speech utterances into segments and clustering these segments into subword categories. In previous approaches, the clustering step usually assumes the number of subword units are known beforehand, which is unreasonable for zero-resource languages. Moreover, the previously-used clustering methods are not able to detect non-spherical clusters that are often present in real-world speech data. We address the two problems by a brand new clustering method, called density peak clustering (DPC), which is motivated by the observation that cluster centers are characterized by a higher density than their neighbors and by a relatively large distance from other points of a higher density in the space. Experiments on unsupervised acoustic units discovery demonstrate that our DPC approach can easily discover the number of subword units and it outperforms the recently proposed normalized cuts (NC) clustering approaches [1].**

I. Introduction

Unsupervised acoustic modeling has drawn much attention recently with the tremendous growth of online speech data and urgent needs for low-resource speech processing. Traditional acoustic modeling has been using a highly supervised paradigm that needs a large set of labelled speech data and language-specific linguistic knowledges about phoneme inventory and pronunciation lexicon. However, labelled data and human expertise are usually unavailable for low- or zero-resource languages and even for some major languages, because data annotation takes tremendous manual effort. With the exponential growth of cheap, online speech data, unsupervised speech modeling with less human effort has drawn much research interest lately. Some recent promising techniques have been successfully applied to a variety of applications, including speech recognition [2], spoken term detection [3] [4], topic segmentation [5] and classification [6] [7] [8].

A fundamental issue for unsupervised acoustic modeling is to discover the *subwords*, e.g., phoneme-like units, automatically from speech in the given language. After that, each discovered subword cluster can be modeled by a Hidden Markov Model (HMM)[9] using traditional supervised means. Inspired by an early acoustic segment modeling (ASM) [10] [11] approach in isolated word recognition, unsupervised subword unit discovery is usually approached by a *segmentation-clustering* strategy. First, we segment the running speech in an unsupervised manner, also known as phoneme/acoustic-unit segmentation, which aims to discover boundaries between phoneme-like units from a speech feature stream according to changes in acoustic characteristics [12] [13]. Second, we cluster the resulting segments of variable lengths into clusters, each of which is considered as an acoustic unit. Subsequently, for each speech utterance, cluster membership labels are used as transcriptions for HMM-based acoustic modeling. With the cluster labels as an initial bootstrap, the acoustic models can be further refined via an iterative training process [14], in which model parameter estimation and utterance decoding are performed alternately. Segmentation, clustering and acoustic modeling are often considered separately. It is also possible to consider them simultaneously under a nonparametric Bayesian approach [15].

Vector quantization (VQ) is a straightforward approach to segment clustering, in which k-means clustering is performed on the mean vector of the frame-level spectral feature in each segment [10] [16] [17]. In order to exploit the trajectory dynamics of speech, segmental GMMs (SGMMs) have been introduced to represent segments with a polynomial function and segment representations are fitted into a mixture model for clustering purpose [7] [18]. GMM labeling is another approach with two steps. First, a GMM is trained on the frame-level features. Second, the GMM is used as a tokenizer to label a segment.

Compared with spectral features, Gaussian posterior features have shown robustness in speech recognition [19] and spoken term detection [20]. Therefore, Wang *et. al.* have proposed a clustering approach recently with segment-level Gaussian posteriorgram representation [1] [21] [22]. First, they generate segment posteriorgrams by averaging frame-level Gaussian posterior probabilities in each segment. After that, they construct a Gaussian-by-segment matrix by stacking together the segment-level Gaussian posteriorgrams of all utterances. Acoustic unit categories are then discovered by clustering either on the Gaussian components (Gaussian component clustering, GCC) or on the segments (segment clustering, SC). Specifically, they use normalized cuts (NC) as the clustering algorithm: First, a dimensionality reduction algorithm, Laplacian eigenmaps [23] [24], is used to project the big matrix to a low-dimensional matrix while preserving

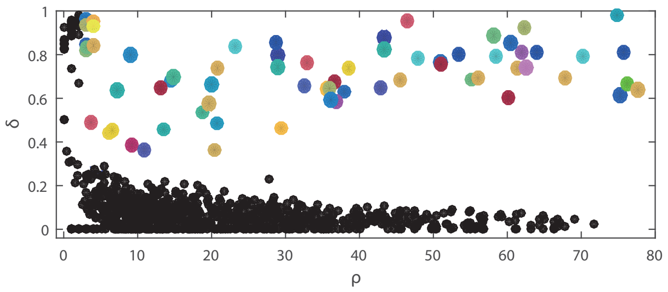


Fig. 1. Decision graph for segments in TIMIT training set.

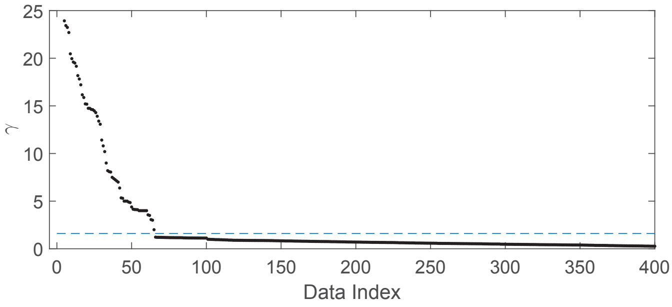


Fig. 2. The value of $\gamma_i = \rho_i \delta_i$ in decreasing order

its locality property; Second, k-means clustering is simply performed on the matrix. The approach achieves state-of-the-art performance that consistently outperforms VQ and GMM labeling by a large margin.

Previous clustering methods used in acoustic units discovery usually assume that the number of subword units are known beforehand. For example, in Wang's NC approach [1], the number of subword units is set to the phoneme number of the language under consideration. However, phoneme number is usually not known for zero-resource languages or surprise language scenarios. Our study in this paper shows that the acoustic unit discovery performance is highly affected by the pre-set number of clusters. On the other hand, k-means is often used as the clustering algorithm in previous approaches, e.g., VQ [16] [10] and NC [25] [26]. However in k-means, because a data point is always assigned to the nearest center, it is not able to detect nonspherical clusters (or arbitrary-shaped clusters) that usually present in real-world speech data.

In this paper, we address the two problems by a new clustering method called *density peak clustering* (DPC) [27]. Different from k-means that solely relies on distance, DPC is based on the idea that cluster centers are characterized by a higher density than their neighbors and by a relatively large distance from points with higher densities. Thus clusters are robustly recognized regardless of their shapes. Besides, this clustering algorithm is able to automatically find the number of clusters. Subword unit discovery experiments on TIMIT demonstrate that our DPC approach apparently outperforms VQ [10] [17] and NC [21].

II. Clustering by Finding Density Peaks

In this paper, we use a recently proposed clustering method called *density peak clustering* (DPC) [27] for unsupervised acoustic units discovery. Different from commonly used k-means [28] and k-medoids [29] which always assign a data point to the nearest center, DPC performs clustering by finding density peaks. The intuitive idea is that cluster centers always have a higher density than their neighbors and cluster centers are separated by a large distance. The clustering procedure makes the number of clusters arising intuitively, outliers are automatically spotted and excluded, and clusters are detected regardless of their shape.

For each data point i to be clustered, we define ρ_i as its local density and δ_i as its distance from points of higher density. The former is defined as

$$\rho_i = \sum_j \chi(d_{ij} - d_c) \tag{1}$$

where $\chi(x) = 1$ if $x < 0$ and $\chi(x) = 0$ otherwise. Here d_{ij} is a distance metric between point i and j and d_c is a cutoff distance. We can clearly see that ρ_i is equal to the number of points closer than d_c to point i. δ_i is the minimum distance between the point i and any other point with higher density:

$$\delta_i = \min_{j:\rho_j > \rho_i} (d_{ij}). \tag{2}$$

For the point with highest density, we take $\delta_i = \max_j d_{ij}$. The points with local or global maxima in density have larger δ_i than the nearest data points. Thus, cluster centers are the points with high δ and relatively high ρ, this is the core idea of this algorithm. The only parameter to be set is the cutoff distance d_c. It is proven that the clustering results are robust with respect to the choice of d_c for large data sets [27].

Fig. 1 illustrates the so-called *decision graph* generated by DPC clustering on the segments. Each point in this figure represents a segment that is represented by its segment-level Gaussian posterior representation (see Section 3.1). All the segments in the TIMIT training set are plotted in this figure according to their ρ and δ calculated by Eq. (1) and Eq. (2). The decision graph shows the presence of several distinct density maxima while the exact number is not clear. A hint for choosing the number of centers is provided by $\gamma_i = \rho_i \delta_i$ sorted in decrease order. As shown in Fig. 2, γ_i starts to grow drastically below data index 65. Therefore, 65 is selected as the number of clusters. Back to Fig. 1, we can see that the cluster centers (colored) stand out from the majority of the points (gray). These centers are marked as big colored points that have high δ and relatively high ρ. The number of the cluster center (65) is near the number of phonemes (61) provided by TIMIT.

Fig. 3 shows the flow chart of acoustic subword units discovery. First, we partition each speech utterance into the variable-length segments using an unsupervised segmentation approach based on acoustic similarity [1]. After that, we discover subword unit categories by clustering.

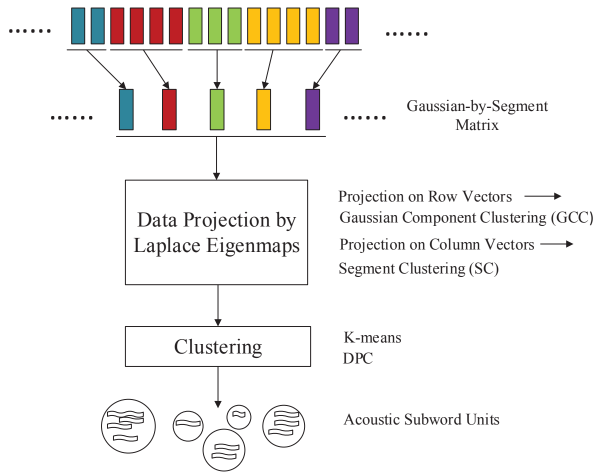

......
......
......
......
Gaussian-by-Segment Matrix
Data Projection by Laplace Eigenmaps
Projection on Row Vectors
Gaussian Component Clustering (GCC)
Projection on Column Vectors
Segment Clustering (SC)
Clustering
K-means
DPC
Acoustic Subword Units

Fig. 3. Flowchart of acoustic subword units discovery.

With the cluster centers, each remaining point is simply assigned to the same cluster as its nearest neighbor of higher density. In contrast with other clustering algorithms (e.g. k-means) in which an objective function is optimized iteratively, the cluster assignment is performed in a single step. Thus it is faster and more efficient than other algorithms.

III. Acoustic Subword Units Discovery

A. *Segment-level Feature Representation*

Before clustering, we need to represent a segment with a segment-level feature representation. A straightforward way is to average the frame-level MFCC vector and result in a mean MFCC representation for each segment. However, GMM posterior is a more robust representation of speech [19] [20]. Suppose a speech utterance is denoted by $\mathbf{O} = [\mathbf{o}_1, \mathbf{o}_2, \cdots, \mathbf{o}_T]$, where $\mathbf{o}_t$ is the MFCC vector of frame t. We use $\{C_1, C_2, \cdots, C_M\}$ to denote the M Gaussian components. The posterior probability vector of frame t is represented by

$$\mathbf{q}_t = [p(C_1|\mathbf{o}_t), p(C_2|\mathbf{o}_t), \cdots, p(C_M|\mathbf{o}_t)]^T. \qquad (3)$$

Hence the Gaussian posteriorgram of utterance $\mathbf{O}$ is defined as

$$\mathbf{Q} = [\mathbf{q}_1, \mathbf{q}_2, \cdots, \mathbf{q}_T]. \qquad (4)$$

The posterior representation for a segment is to simply average the frame-level posterior vectors within the span of each segment, and the segment-level Gaussian posteirorgram of utterance $\mathbf{O}$ is

$$\bar{\mathbf{Q}} = [\bar{\mathbf{q}}_1, \bar{\mathbf{q}}_2, \cdots, \bar{\mathbf{q}}_K], \qquad (5)$$

where K is the number of the segments in the utterance. Fig. 4 shows the segmental-level Gaussian posteriorgram of a speech utterance in the TIMIT training set.

By stacking the segment-level Gaussian posterior representation of all the utterances in a speech dataset, we obtain a Gaussian-by-segment matrix with size of $M \times N$, where M is number of Gaussian mixtures and N is the number of segments in the whole dataset.

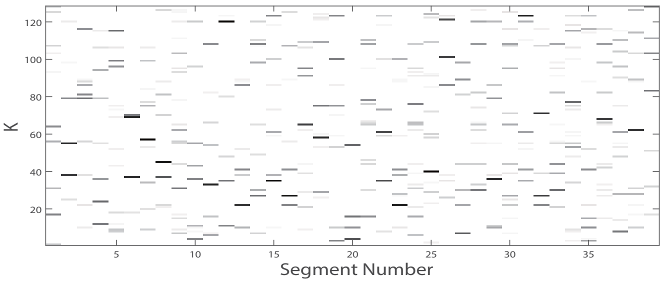


Fig. 4. Segmental-level Gaussian posteriorgram with 128 Gaussian components for an utterance in TIMIT. x-axis: index of segments, y-axis: index of Gaussian components. Darker color means higher Gaussian posterior probability.

Algorithm 1 DPC-GCC

```
Input: Gaussian-by-segment Matrix $\mathbf{X}$
Output: $R$ GMM, cluster membership assigned to each segment
1: Calculate inner-product similarity matrix: $\mathbf{W} = \frac{1}{M}\mathbf{X}\mathbf{X}^T$
2: Calculate the normalized Laplacian matrix: $\mathbf{L} = \mathbf{D}^{-\frac{1}{2}}\mathbf{W}\mathbf{D}^{-\frac{1}{2}}$,
   $\mathbf{D}$ is the diagonal matrix with elements $\mathbf{D}_{ii} = \sum_j \mathbf{W}_{ij}$
3: Derive matrix $\mathbf{U} = [\mathbf{u}_1, \mathbf{u}_2, \cdots, \mathbf{u}_J]$ that contains the eigenvectors
   corresponding to the $J$ largest eigenvalues of $L$
4: Normalize row vectors of $\mathbf{U}$ to have unit L2-norm
5: Perform DPC clustering on the $M$ row vectors of $\mathbf{U}$ and get the cluster
   membership, cluster number is discovered
6: Form a new GMM for each cluster by assigning equal weights to the
   corresponding Gaussian components
7: Label each segment with the index of the GMM that scores the highest
   on it
```

B. *Clustering*

Given the Guassian-by-segment matrix, we can perform clustering either on the row vectors (Gaussian components) or on the column vectors (segments), which lead to Gaussian component clustering (GCC) and segment clustering (SC) respectively. As discussed in [21], the two types of clustering have duality and they both lead to reasonable acoustic unit categories. However, the original Guassian-by-segment matrix is too large. we need to project this matrix to a reasonable size by a dimensionality reduction algorithm. In [1], Laplacian eignmaps is used to this purpose. The flowchart of clustering is illustrated in Fig. 3.

1) Gaussian Component Clustering by DPC: In [1], the NC-GCC clustering is divided into two consecutive steps: calculation of the Laplacian eigenvectors and k-means clustering. In this paper, we substitute k-means with DPC in the second step. The DPC-GCC approach is shown in Algorithm 1. The steps are similar to those in NC-GCC but the cluster number does not need to be set because DPC can automatically discover it. Given the Gaussian-by-segment matrix $\mathbf{X}$, we first form a similarity matrix $\mathbf{W}$ using the inner product criterion. Then we compute the normalized Laplacian matrix $\mathbf{L}$ and derive the J eigenvectors with J largest eigenvalues. After that, DPC is applied to row vectors of $\mathbf{U}$. As a result of the clustering, a set of new GMMs are generated and used for segment labeling.

TABLE I
CLUSTERING PERFORMANCE (NMI) OF DIFFERENT GCC APPROACHES.

Number of Gaussians: M	128	256	512	768	1024	2048	3072	4096
NC-GCC [1]	0.299	0.305	0.313	0.328	0.336	0.342	0.347	0.348
DPC-GCC	**0.303**	**0.311**	**0.318**	**0.329**	**0.340**	**0.352**	**0.359**	**0.361**

Algorithm 2 DPC-SC

```
Input: Gaussian-by-segment Matrix $\mathbf{X}$
Output: Cluster membership assigned to each segment
1: Calculate vector $\mathbf{d} = \mathbf{X}^T(\mathbf{X}\mathbf{1})$, where $\mathbf{1}$ is the unit vector. Let $\mathbf{D}$ be the diagonal matrix with $\mathbf{d}$ on its diagonal position
2: Transform $\mathbf{X}$ to $\tilde{\mathbf{X}} = \mathbf{X}\mathbf{D}^{-\frac{1}{2}}$
3: Calculate similarity matrix of row vectors: $\tilde{\mathbf{W}} = \tilde{\mathbf{X}}\tilde{\mathbf{X}}^T$
4: Derive matrix $\tilde{\mathbf{U}} = [\tilde{\mathbf{u}}_1, \tilde{\mathbf{u}}_2, \cdots, \tilde{\mathbf{u}}_J]$ that contains the eigenvectors corresponding to the $J$ largest eigenvalues of $\tilde{\mathbf{W}}$
5: Comput $\mathbf{U} = \tilde{\mathbf{X}}^T\tilde{\mathbf{U}}$
6: Normalize row vectors of $\mathbf{U}$ to have unit L2-norm
7: Perform DPC clustering on the $N$ row vectors of $\mathbf{U}$ and get the cluster memberships of the segments, cluster number is discovered
```

2) Segment Clustering by DPC: Similarly, clustering can be performed on the column vectors, resulting in segment clustering (SC). The clustering procedure is summarized in Algorithm 2. The number of segments N can be quite large in practice and directly computing segment similarity matrix is difficult because of the huge memory cost. Again, as suggested in [21], we derive the eigenvector from matrix $\tilde{\mathbf{U}}$ in step 4 of Algorithm 2 avoiding directly computing Laplace matrix $\mathbf{L}$. It has been proven that $\mathbf{U}$ consists of the J eigenvectors of $\mathbf{L} = \mathbf{D}^{-\frac{1}{2}}\mathbf{W}\mathbf{D}^{-\frac{1}{2}}$. Refer to [23] for more details.

IV. Experiments

A. Experimental Setup

We carried out experiments on the training part of the TIMIT corpus [30] that contains a total of 4620 sentences from 462 speakers. The 39-D MFCC vectors were post-processed by mean and variance normalization (MVN) and vocal tract length normalization (VTLN). There are a large number of silence segments in the corpus and they may bias the clustering results. Thus we remove the silence segments according to the manual transcripts provided by the corpus. We partitioned each utterance into segments with variable-length using a simple but effective method introduced in [21]. According to [27], clustering results are robust with respect to the choice of d_c and one can choose it so that the average number of neighbors is around 1% to 2% of the total number of points. We set this percentage to 2% instead of directly setting the value of d_c. In Algorithm 1 and Algorithm 2, J determines the reduction degree of the data and we set J to 70 empirically according to [23]. The clustering performance was evaluated with reference to frame-level manual phoneme labels. A commonly used metric for the clustering task, normalized mutual information (NMI), was used as performance metric. A larger NMI value indicates a better clustering result. We compared the proposed DPC approach with the NC approach [1]. The number of clusters was set to 61, i.e., the real phoneme number in the TIMIT set, in the NC experiments. In the DPC experiments, the distance metric d is cosine.

TABLE II
CLUSTERING PERFORMANCE OF DIFFERENT SC APPROACHES.

Approach	NMI
VQ (MFCC)	0.317
DPC-SC (MFCC)	**0.338**
NC-SC [1]	0.351
DPC-SC	**0.368**
NC-SC (w/ Iterative Training) [1]	0.391
DPC-SC (w/ Iterative Training)	**0.412**

B. Results on GCC

Table I shows the clustering performances of different GCC approaches. We tested different number of Gaussian components (M from 128 to 4096) in the GMM posterior feature representation. Firstly, we observe that the performance improves with the increase of Gaussian components. Secondly, the proposed DPC-GCC approach apparently outperforms the NC-GCC approach at all numbers of Gaussian components (the differences are significant at $p < 0.05$ [31]). With the help of the Laplacian representation, our DPC approach can achieve the best NMI of 0.361 (M=4096).

C. Results on SC

Segment clustering (SC) results are shown in Table II. Here we show results on both MFCC and Gaussian posterior representations, where M is set to 4096. We directly perform clustering on MFCC-by-segment matrix for sanity check. As expected, DPC-SC performs significantly better than VQ. This indicates that density peak clustering outperforms k-means in acoustic unit discovery. When Gaussian posterior is used as segment representation, clustering performance is lifted to a new level. Again, our DPC-SC approach clearly outperforms the NC-SC approach (significant at $p < 0.01$). We also notice that the DPC-SC approach has slightly better NMI (0.368) than that of DPC-GCC (0.361). Finally when iterative training [14] is used, the results are further improved and the DPC-SC approach is still the better one with the highest NMI of 0.412. Please also bear in mind, as compared with NC, the DPC approach does not need the input of the cluster number.

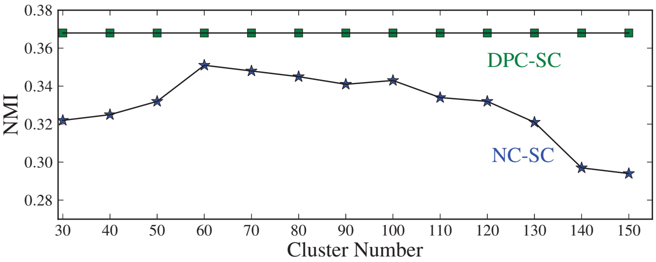
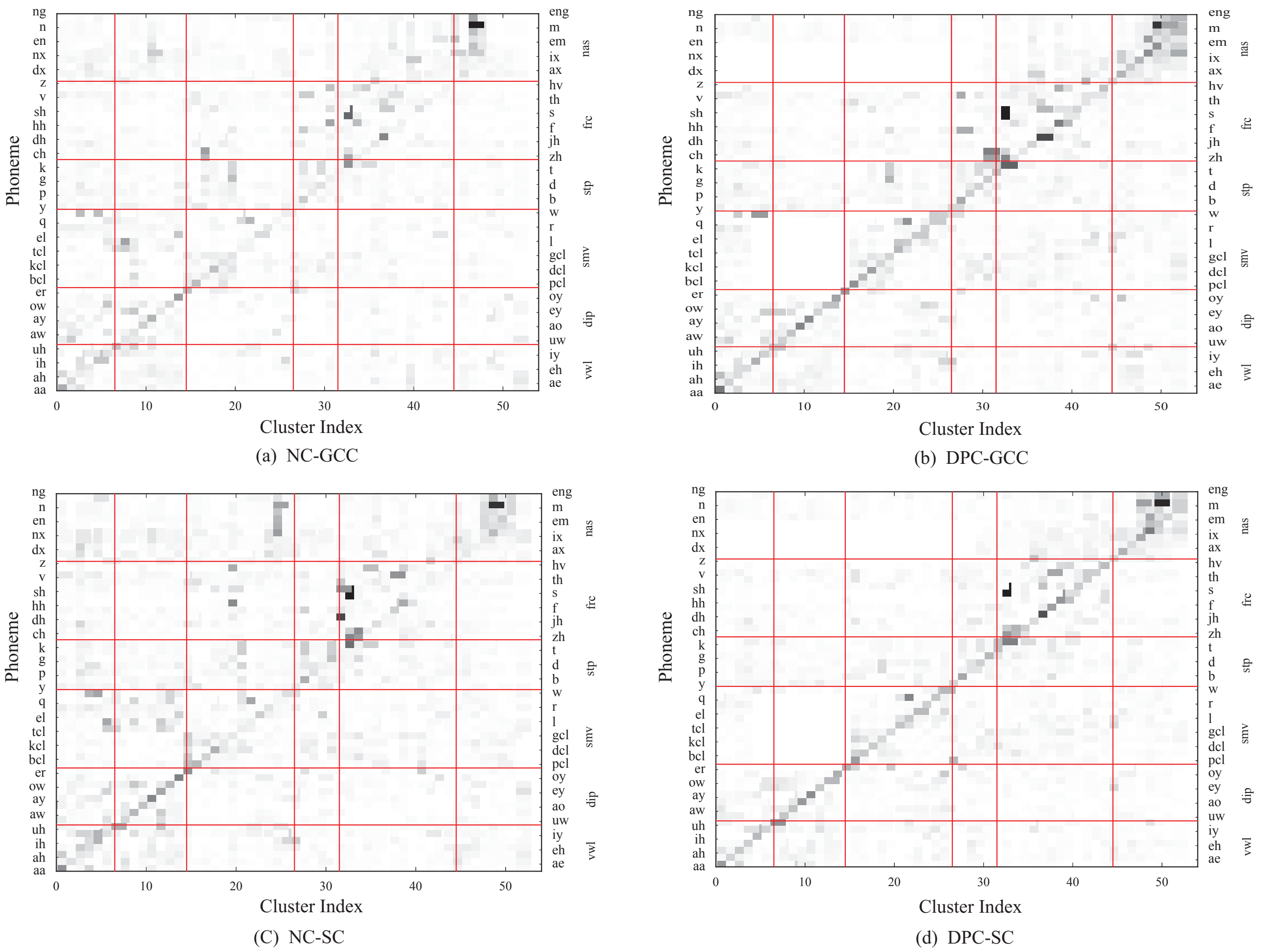


Fig. 5. The NMI of SC-NC when different cluster number is set. The NMI of DPC-NC is also shown for comparison while cluster number is automatically derived.



(a) NC-GCC

(b) DPC-GCC

(C) NC-SC

(d) DPC-SC

Fig. 6. Confusion matrixes between standard phonemes and cluster indexes on TIMIT training set. The darkness is scaled to the value of the corresponding element in the confusion matrix.

D. Analysis

In order to show how the number of clusters affects the performance of NC-SC, we run experiments and draw Fig. 5. We can see that when the true cluster number is set, the performance is the best. However, if this number is shifted away from the true value, NMI degrades dramatically. The performance of NC-SC never reaches the level of DPC-SC that automatically discover the number of clusters. Fig. 6 shows the confusion matrixes between true phonemes and the subword unit clusters discovered. We can observe a diagonal line in each matrix, which means the discovered subword units have clear correlations with phonemes. When we compare NC with DPC, i.e., (a) vs. (b) and (c) vs. (d) in Fig. 6, we see more salient diagonal lines in the confusion matrixes made by DPC. This observation also confirms that DPC has better clustering performance. However, in the nasal broad category (labeled as nas in Fig. 6), significant inter-class confusion appears between phonemes ng, n, en, eng, m and em. This means these nasals are difficult to discriminate by current clustering algorithms.

V. Conclusions and Future Work

This paper investigated automatic subword units discovery from speech in an unsupervised manner. Specifically, we addressed two important problems that have been neglected by previous approaches: the number of clusters has to be determined in advance and the clustering algorithm lacks of robustness in detecting nonspherical clusters. These two problems affect the practical use and robustness of unsupervised acoustic unit discovery. In this paper, we alleviated the two problems by a new clustering algorithm called density peak clustering (DPC). Experiments show that our DPC approach can easily discover the number of subword units and it outperforms the recently proposed normalized cuts (NC) clustering approach [1]. In future work, we plan to test our approach on more challenging low-resource corpora and multilingual conditions.

VI. Acknowledgements

We would like to thank Dr. Haipeng Wang, Dr. Cheung-Chi Leung and Dr. James Glass for valuable discussions. This work is supported by the National Natural Science Foundation of China (Grant No. 61175018, 61571363) and the DSO funded project MAISON DSOCL14045.

References

[1] H. Wang, T. Lee, C.-C. Leung, B. Ma, and H. Li, "An acoustic segment modeling approach to query-by-example spoken term detection," in *I-CASSP 2012 – Annual Conference of the IEEE International Conference on Acoustics, Speech, and Signal Processing(ICASSP), March 25–30 , Kyoto, Japan, Proceedings*, 2012, pp. 5157–5160.

[2] C.-Y. Lee, Y. Zhang, and J. Glass, "Joint learning of phonetic units and word pronunciations for asr." in *EMNLP 2013 –International Conference on Empirical Methods in Natural Language Processing, October 18–21, Seattle, USA, Proceedings*, 2013, pp. 182–192.

[3] A. Muscariello, G. Gravier, and F. Bimbot, "Zero-resource audio-only spoken term detection based on a combination of template matching techniques," in *INTERSPEECH 2011 – 12th Annual Conference of the International Speech Communication Association, August 27–31, Florence, Italy, Proceedings*, 2011.

[4] H. Wang, T. Lee, C.-C. Leung, B. Ma, and H. Li, "Using parallel tokenizers with dtw matrix combination for low-resource spoken term detection," in *ICASSP 2013 – Annual Conference of the IEEE International Conference on Acoustics, Speech, and Signal Processing(ICASSP), May 26–31 , British Columbia, Canada, Proceedings*, 2013, pp. 8545–8549.

[5] I. Malioutov, A. Parkand, R. Barzilay, and J. Glass, "Making sense of sound: Unsupervised topic segmentation over acoustic input," in *ACM 2007 – 45th Annual Meeting of the Association for Computational Linguistics, June 25–27, Prague, Czech Republic, Proceedings*, 2007, p. 504.

[6] H. Gish, M.-H. Siu, A. Chan, and B. Belfield, "Unsupervised training of an hmm-based speech recognizer for topic classification," in *INTERSPEECH 2010 – 11th Annual Conference of the International Speech Communication Association, September 26–30 , Chiba, Japan, Proceedings*, 2010.

[7] M. Siu, H. Gish, A. Chan, and W. Belfield, "Improved topic classification and keyword discovery using an hmm-based speech recognizer trained without supervision," in *INTERSPEECH 2010 – 11th Annual Conference of the International Speech Communication Association, September 26–30 , Chiba, Japan, Proceedings*, 2010.

[8] M. Dredze, A. Jansen, G. Coppersmith, and K. Church, "Nlp on spoken documents without asr," in *EMNLP 2010 –International Conference on Empirical Methods in Natural Language Processing, October 9–11, Massachusetts, USA, Proceedings*, 2010, pp. 460–470.

[9] C. Chan and L. Lee, "Unsupervised hidden markov modeling of spoken queries for spoken term detection without speech recognition," in *INTERSPEECH 2012 – 13th Annual Conference of the International Speech Communication Association, September 9–13 , Oregon, USA, Proceedings*, 2012.

[10] C. Lee, F.Soong, and B. Juang, "A segment model based approach to speech recognition," in *ICASSP 1988 – Annual Conference of the IEEE International Conference on Acoustics, Speech, and Signal Processing(ICASSP), April 11–14, New York, Proceedings*, 1988, pp. 501–541.

[11] G. L. Sarada, A. Lakshmi, H. A. Murthy, and T. Nagarajan, "Automatic transcription of continuous speech into syllable-like units for indian languages," *Sadhana*, vol. 34, no. 2, pp. 221–233, 2009.

[12] Y. Qiao, N. Shimomura, and N. Minematsu, "Unsupervised optimal phoneme segmentation: Objectives, algorithm and comparisons," in *I-CASSP 2008 – Annual Conference of the IEEE International Conference on Acoustics, Speech, and Signal Processing(ICASSP), March 30–April 4, Las Vegas, Nevada, Proceedings*, 2008, pp. 3989–3992.

[13] O. Scharenborg, V. Wan, and M. Ernestus, "Unsupervised speech segmentation: An analysis of the hypothesized phone boundaries," *Acoustical Society of America*, vol. 127, no. 2, pp. 1084–1095, 2010.

[14] H. Wang, T. Lee, C.-C. Leung, B. Ma, and H. Li, "Acoustic segment modeling with spectral clustering methods," *IEEE Transaction on Audio, Speech, and Language Processing*, vol. 23, pp. 264–277, 2015.

[15] C. Lee and J. Glass, "A nonparametric bayesian approach to acoustic model discovery," *Proceedings of the 50th Annual Meeting of the Association for Computational Linguistics*, vol. 1, pp. 40–49, 2012.

[16] B. Ma, D. Zhu, and H. Li, "Acoustic segment modeling for speaker recognition," in *ICME 2009 –Automatic Speech Recognition & Understanding, ASRU. IEEE Workshop , Proceedings*, 2009, pp. 1668–1671.

[17] J. Reed and C. Lee, "A study on music genre classification based on universal acoustic models." in *ISMIR 2006 – 7th International Conference on Music Information Retrieval, Proceedings*, 2006, pp. 89–94.

[18] H. Gish and K. Ng, "A segmental speech model with applications to word spotting," in *ICASSP 1993 – Annual Conference of the IEEE International Conference on Acoustics, Speech, and Signal Processing(ICASSP), April 27–30, Minneapolis, Minnesota, Proceedings*, 1993, pp. 447–450.

[19] G. Aradilla, J. Vepa, and H. Bourlard, "Using posterior-based features in template matching for speech recognition," in *ICSLP 2006 – International Conference on Spoken Language Processing, Proceedings*, 2006.

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

[21] H. Wang, T. Lee, C.-C. Leung, B. Ma, and H. Li, "Unsupervised mining of acoustic subword units with segment-level gaussian posteriorgrams," in *INTERSPEECH 2013 – 14th Annual Conference of the International Speech Communication Association, August 25–29, Lyon, France, Proceedings*, 2013, pp. 2297–2301.

[22] H. Wang, T. Lee, C.-C. Leung, B. Ma, and H. Li, "A graph-based gaussian component clustering approach to unsupervised acoustic modeling," in *INTERSPEECH 2014 – 15th Annual Conference of the International Speech Communication Association, September 14–18, Singapore, Proceedings*, 2014.

[23] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural computation*, vol. 15, no. 6, pp. 1373–1396, 2003.

[24] L. Xie, L.-L. Zheng, Z. Liu, and Y. Zhang, "Laplacian eigenmaps for automatic story segmentation of broadcast news," *IEEE Transaction on Audio, Speech, and Language Processing*, vol. 20, no. 1, pp. 276–289, 2012.

[25] I. Dhillon, Y. Guan, and B. Kulis, "Kernel k-means: spectral clustering and normalized cuts," in *ACM 2004 – 45th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, August 22–25, Seattle, WA, Proceedings*, 2004, pp. 551–556.

[26] J. Shi and J. Malik, "Normalized cuts and image segmentation," *Pattern Analysis and Machine Intelligence*, vol. 22, no. 8, pp. 888–905, 2000.

[27] A. Rodriguez and A. Laio, "Clustering by fast search and find of density peaks," *Science*, vol. 344, no. 6191, pp. 1492–1496, 2014.

[28] J. MacQueen, "Some methods for classification and analysis of multivariate observations," *Berkeley symposium on mathematical statistics and probability*, pp. 281–297, 1967.

[29] L. Kaufman and P. Rousseeuw, *Finding groups in data: an introduction to cluster analysis*. John Wiley & Sons, 2009, vol. 344.

[30] J. Garofolo, L. Lamel, W. Fisher, J. Fiscus, D. Pallett, N. Dahlgren, and V. Zu, "Acoustic-phonetic continuous speech corpus (timit)," *CD-ROM, - NTIS order no. PB91-505065,*, 1991.

[31] P.Koehn, "Statistical significance tests for machine translation evaluation." in *EMNLP 2004 –International Conference on Empirical Methods in Natural Language Processing, July 25–26, Barcelona, Spain, Proceedings*, 2004, pp. 388–395.