Odyssey 2020 The Speaker and Language Recognition Workshop
1-5 November 2020, Tokyo, Japan

ON EARLY-STOP CLUSTERING FOR SPEAKER DIARIZATION

Liping Chen[1], *Kong Aik Lee*[2], *Lei He*[1], *Frank K. Soong*[1]

[1] Microsoft China, Beijing, China
[2] Biometrics Research Laboratories, NEC Corporation, Japan

lipch@microsoft.com, kongaik.lee@nec.com, helei@microsoft.com, frankkps@microsoft.com

Abstract

We propose an early-stop strategy for improving the performance of speaker diarization, based upon agglomerative hierarchical clustering (AHC). The strategy generates more clusters than the anticipated number of speakers. Based on these initial clusters, an exhaustive search is used to find the best possible combination of clusters. The resultant clusters are more homogeneous in their speaker purity, i.e., with less speech of other speakers being merged into the clusters. Experiments on First DIHARD shows that the early-stop clustering and the speaker cluster selection lead to improved cluster purity of speaker and better diarization than the conventional AHC. Moreover, for the case of unknown number of speakers, the proposed approach can estimate the number of speakers more accurately, generate better clusters to the corresponding speakers, and yield a fairly stable system performance against a wide range of stopping threshold.

1. Introduction

Speaker diarization aims to solve the problem of "*who spoke when?*" given an audio recording. It is applicable, for instance, in profiling and analyzing the interaction among attendees in a meeting (e.g., who is the dominant speaker?), and as the essential processing front-end to automatic speech recognition with multiple speakers in the recordings [1]. Speaker diarization has, therefore, attracted a great deal of interest in the past decades. Research in this direction has gained further momentum recently driven by public challenges, like the *DIHARD speaker diarization challenge* [2, 3, 4] and the *multi-speaker detection task* in the recent NIST SRE'18 [5] and SRE'19 [6] *speaker recognition evaluation*.

In its most common setup, speaker diarization is accomplished with the following steps: *temporal segmentation*, *clustering*, followed by *temporal re-alignment* [1, 7]. In the temporal segmentation step, an input recording is split into homogeneous segments with, ideally, one speaker per segment. This is typically accomplished by detecting the change points between speakers in the recording. Segments are then formed by taking the interval between two change points. In [8, 9], a uniform segmentation scheme (e.g., 1 to 2 seconds) has shown to work well for multi-speaker detection task. In the clustering stage that follows, segments belonging to the same speaker are merged, such that the number of clusters corresponds to the number of speakers (estimated or given a priori) in the recording. In this regard, agglomerative hierarchical clustering (AHC) is commonly used. In [8] and [9], segments are first represented as fixed-length vectors, i.e., speaker embedding (e.g., i-vector or x-vector), before AHC is applied. In [10], an additional *variational Bayesian* (VB) clustering step is applied after AHC. In the final step, temporal re-alignment is performed at a finer granularity where speech frames are re-aligned to speaker states. This is accomplished with a *hidden Markov model* (HMM) [11, 12] where each state corresponds to a speaker. Since the HMM is initialized with the results from the clustering step, a good clustering result benefits substantially the overall system performance.

The number of clusters and therefore the number of HMM states is associated with the number of speakers in a speech recording. This number has to be estimated if not known a priory. Accurate estimation of the number of speakers has been a long-standing challenge in speaker diarization research. In [8, 9], speaker embeddings were used in conjunction with probabilistic linear discriminant analysis (PLDA) to derive a score matrix for AHC. The number of speakers is determined with a threshold set on the similarity scores from the merged clusters. Similarly, in the diarization system based on variational Bayes HMM (VB-HMM), the number of speakers was decided by the resultant number of states in the HMM [13].

In the AHC speaker diarization framework, the number of speakers equals the number of clusters when the clustering stops. During the clustering process, speech frames from interfering speakers may get merged due to two factors. One is that there may be missed speaker change points in some segments, i.e., the speech segments with more than one speakers may get merged. The importance of speaker change detection accuracy on diarzation performance has been verified in prior work like [14] and [15]. The other is that the speaker modeling and scoring techniques cannot avoid the speech segments from interfering speakers being merged. The existence of interfering speaker will bias the cluster of the target speaker (whose speech dominates the cluster). The worst case is that at the end of clustering, the clusters of the target and interfering speakers get merged. Such clustering methods, where the segments are clustered into the number of speakers, heavily depend on the speech segmentation, the speaker modeling and similarity scoring. To this end, some research work has been carried out on the cluster purification during and after clustering like [16], [17] and [18]. In [16], the purity problem was categorized into two types, i.e., segment level and frame level. The crux of the purification methods lie in detecting the segments from interfering speakers and the silence in the clusters obtained by AHC.

In this paper, we aim to reduce the effects of the missed speaker change points, the speaker modeling and similarity measurement in speaker clustering. By doing so, we improve the cluster purity to initialize the state emission probabilities in the subsequent Viterbi re-segmentation. The core idea in our work is to insert an early-stop criterion into the AHC to obtain more clusters than the anticipated number of speakers. The rationale is that, with more clusters, we reduce the chance of speech frames from interfering speakers being merged to those pertaining to other speakers. The idea of stopping early is not new to speaker diarization. For example, in [19], the iterations

10.21437/Odyssey.2020-16

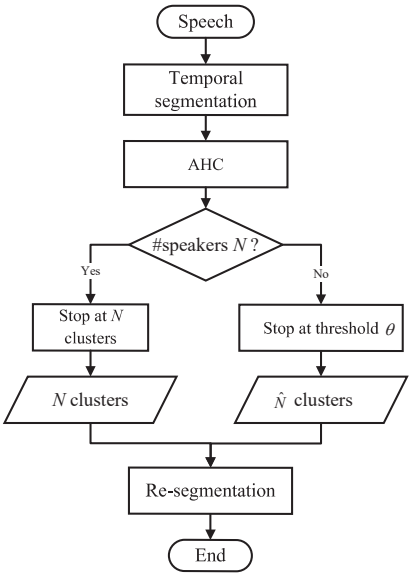

Figure 1: Flowchart of the diarization system based on agglomerative hierarchical clustering.

for Viterbi re-segmentation were stopped early before convergence. In our work, we demonstrate the early-stop strategy in AHC; build a diarization system based on that and illustrate the early-stop AHC systematically. Both cases where the number of speakers is given or has to be estimated are included.

The remainder of the paper is organized as follows. In Section 2, we present a brief overview of the speaker diarization frameworks based on the agglomerative hierarchical clustering. Our proposed method on early-stop clustering combined with the number of speakers estimation and cluster selection is presented in Section 3. The experiments will be presented in Section 4. Finally, Section 5 concludes the paper.

2. Speaker diarization framework

In this section, we describe briefly the *agglomerative hierarchical clustering* (AHC) and re-segmentation methods used in this paper.

2.1. Agglomerative hierarchical clustering

The AHC process begins with temporal segmentation. As shown in Fig. 1, the input speech recording is split into variable-length segments with the help of a speaker change point detector, or uniform segments of fixed duration. These segments are merged based on their similarity measure until a stopping criterion is met. In the case where the number of speakers N is given, cluster merging stops when the number of clusters N is attained. Alternatively, cluster merging could be stopped based on the threshold θ on the speaker similarity. The number of clusters $\hat{N}$ is taken as the number of speakers. In both cases, it is assumed that each cluster corresponds to a speaker in the input recording.

At the core of AHC is the similarity matrix consisting of the affinity measures between segments. Two methods that have shown to be effective are: (i.) speaker embedding followed by cosine or PLDA scoring, (ii.) *Bayesian information criterion* (BIC) [20]. In the first method, segments are first represented as fixed-length vectors (e.g., x-vector). The affinity between two segments is given by the cosine similarity (or PLDA score) between two embeddings. In the BIC method, the affinity measure is given by the log-likelihood ratio between a mono-Gaussian (same speaker) and a bi-Gaussian (different speakers) hypotheses, which indicates whether two segments should be merged or remains as two separate clusters.

2.2. Re-segmentation

The results from the AHC gives a rough estimate of diarization hypothesis at the segment level. Let $\{\mathcal{C}_1, ..., \mathcal{C}_N\}$ denotes the clustering result from the AHC stage, where N indicates the number of target speakers in the recording and $\mathcal{C}_n$ denotes the segments in the n-th cluster. This cluster hypothesis is used to initialize a HMM consisting of N states. The emission probability distribution of each state is represented as a GMM. The Viterbi alignment is used to assign speech frames to the N states. The process is repeated for a considerable number of iterations. In our proposed early-stop clustering (Section 3), the N clusters corresponding to the speakers are selected from a set of K clusters ($K \geq N$). Re-segmentation will play two roles: 1. re-aligning of frames among the N clusters; 2. aligning of frames in the remaining $K - N$ clusters to the N clusters.

Re-segmentation is performed using speaker bottleneck features instead of *mel-frequency cepstral coefficients* (MFCC) features. The speaker bottleneck features are extracted with a *deep neural network* (DNN) trained to discriminate speakers [21]. We use two iterations in this paper since we didn't observe significant difference with more iterations. We apply a flat state transition probabilities for all states as the speaker bottleneck feature is supposed to be dominated by speaker instead of phonetic attributes and every speaker should be equally treated.

3. Early-stop clustering

This section describes the early-stop clustering strategy. The central idea of which is to produce more clusters than the anticipated number of speakers in a recording.

3.1. Early-stop strategy

Let $K \geq N$ be a number greater than the actual number of speakers N in a given recording. The aim of early-stop clustering is to improve the purity of the resulting clusters by stopping the AHC cluster merging at an earlier stage with a stricter threshold. The rationale for early-stop clustering is two-fold. Firstly, speaker change point detection is never perfect, let alone cases when uniform segmentation is used. An early-stop strategy reduces the chances of those segments with more than one speakers being merged to other clusters. Secondly, speaker discrimination is never perfect. Even for the case with deep speaker embedding and cosine scoring back-end, segments belonging to different speakers might be merged into the same cluster due to higher similarity in terms of phonetic content, background acoustic, etc.

Fig. 2 shows the speaker diarization system with early-stop clustering. The threshold on the speaker similarity is denoted as θ_{s}, with the subscript s denoting *strict*. Similar to the AHC threshold θ in Fig. 1, the threshold θ_{s} is determined on a development set. As we shall show in the experiment, there is a wide range of values for θ_{s} that give good diarization performance. With more clusters (and higher purity) than required, our next step is *cluster selection*. In the case where the number of speakers N is known, N clusters will be selected from the K clusters. For the case when N is unknown, the number of speakers $\hat{N}$ is estimated first.

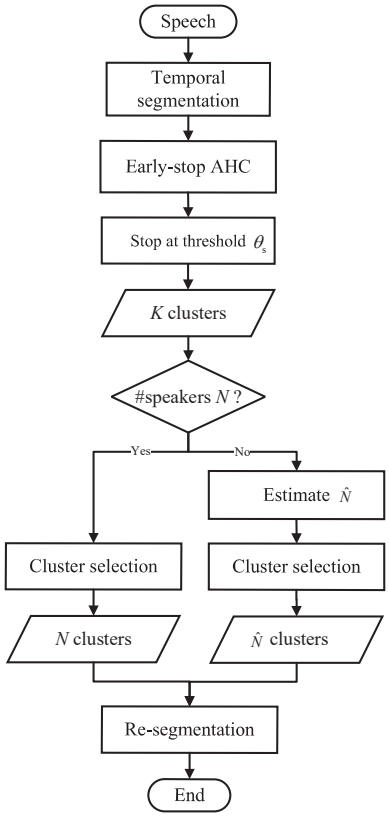

Figure 2: Flowchart of the speaker diarization system with early-stop clustering.

Number of speakers estimation. Given the K clusters from the early-stop clustering stage, we use a simple strategy to estimate the number of speakers based on the *maximum eigen-gap* method, detailed as follows.

i. *Similarity matrix* – Compute the similarity matrix $\mathbf{S} = [S_{j,k}]$ among the K clusters with $S_{j,k}$, for $j, k = 1, ..., K$, denotes the affinity measure between clusters j and k. With x-vector embeddings, $S_{j,k}$ are the cosine scores between clusters. Alternatively, the BIC scores (as described in Section 2) computed on the frames in the clusters could be used.

ii. *Spectral eigen-ratio* – Let $\{e_1, ..., e_K\}$ be the eigenvalues of the similarity matrix $\mathbf{S}$ sorted in descending order. We define the eigen-ratio between two successive eigenvalues as $\varsigma_k = e_k / e_{k+1}$, for $k = 1, 2, ..., K-1$.

iii. *Maximum eigen-gap* – We estimate the number of speakers, $\hat{N}$, as the index k that gives the largest eigen-gap between two successive eigenvalues:

$$\hat{N} = \arg\max_k \varsigma_k \tag{1}$$

3.2. Cluster selection

The *early-stop clustering* strategy aims to produce cluster hypothesis that truly represent individual speakers in terms of cluster purity. In the following, we use the notation N to denote the number of speakers (as a surrogate to the estimate $\hat{N}$) for brevity. Given $K \geq N$ clusters, cluster selection is performed in an exhaustive search manner as follows:

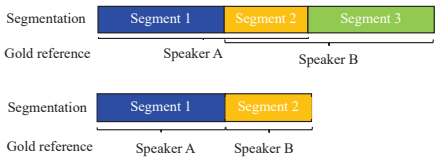

Figure 3: Segmentation according to the gold reference.

i. *Exhaustive cluster subsets* – Randomly select N clusters among the K clusters. Let $I_l = \{i_1, ..., i_n, ..., i_N\}$ be the l-th index combination, where i_n indicates the index of the n-th selected cluster in the K original clusters. Denote the set of index combinations to be $\{I_1, ...I_l, ...I_L\}$ which covers all the combinations of N cluster indices from the K clusters. L is the number of combinations. Given the l-th $(l = 1, ..., L)$ combination I_l, a cluster subset $\{\mathcal{C}_{I_l}\}$ can be obtained by selecting the clusters from the K clusters indexed by I_l.

ii. *Score sub-matrices* – For the i-th cluster subset, we extract the corresponding rows and columns from $\mathbf{S}$ to form its score sub-matrix as $\mathbf{s}_l = \mathbf{S}(:, I_l) \bigcap \mathbf{S}(I_l, :)$. The sub-matrix is extracted for all the L cluster subsets, resulting in a set of score sub-matrices as $\bigcup_{l=1}^{L} \mathbf{s}_l$.

iii. *Eigenvalue summation* – Do eigenvalue decomposition on the L sub-matrices and sum up the eigenvalues for each sub-matrix. Let $\{\eta_1, ..., \eta_L\}$ denote the sum of eigenvalues for all the sub-matrices.

iv. *Cluster selection* – Select the I_{l^*} combination that gives the maximum eigenvalue sum:

$$l^* = \arg\max_l \eta_l \tag{2}$$

We select the N clusters as indexed by I_{l^*} from the K clusters, each corresponding to a speaker. The speech frames in the remaining $K - N$ clusters are aligned to the N cluster via Viterbi re-segmentation step as described in Section 2.

4. Experiments

Our experiments on the First DIHARD dataset used the gold speech segmentation as defined for Track 1 according to the evaluation plan [3]. Also, no forgiveness window was used and the overlap speech was evaluated. The conversation files were segmented according to the reference *voice activity detection* (VAD) as shown in Fig 3. Overlap segments were treated as separate segments. All the parameters (clustering stop threshold) were tuned on the development set and then applied to the evaluation set. Diarization Error Rate (DER) was used as the performance metric. Since we did not aim to tackle overlap speech, overlap speech will be assigned to one of the speakers (considered as a miss to other speakers in the overlap segments). The DER was computed with the md-eval tool [3].

For the acoustic feature, we used 80-dimensional log mel-filterbank features. A ResNet [22] x-vector extractor was trained with VoxCeleb 1 and 2 datasets [23, 24] for speaker embedding. There were 7363 speakers with 2794 hours of speech in total. Data augmentation including reverberation and noise addition were exerted randomly on the audio files, which doubled the size of the training set. At the input, frames were concatenated with 40 frames on both left and right sides into 2-D arrays. The structure of the ResNet is shown in Table 1. We had

eight residual convolutional blocks, i.e., conv{2,...,9}. Each block contains two convolutional layers, denoted as _1 and _2, respectively. The shortcut connection from the input to the output within each block is not shown explicitly in the table. The model was trained to discriminate among the speakers in the training set with respect to a softmax output layer and the cross-entropy loss. The linear output of the layer before the output layer whose size was 128 was used as the x-vector. Pytorch was used for model training. The details of the ResNet structure can be found in the online torchvision scripts [1].

Table 1: ResNet architecture used for x-vector extraction. The format in Bolock's Structure are: receptive filed size, number of channels, stride, padding.

Block Name	Block's Structure
conv1	$7 \times 7, 64, (2,2), (3,3)$
max pooling	$3 \times 3, 2, 1$
conv2_1	$3 \times 3, 64, (1,1), (1,1)$
conv2_2	$3 \times 3, 64, (1,1), (1,1)$
conv3_1	$3 \times 3, 64, (1,1), (1,1)$
conv3_2	$3 \times 3, 64, (1,1), (1,1)$
conv4_1	$3 \times 3, 128, (1,1), (1,1)$
conv4_2	$3 \times 3, 128, (1,1), (1,1)$
conv5_1	$3 \times 3, 128, (1,1), (1,1)$
conv5_2	$3 \times 3, 128, (1,1), (1,1)$
conv6_1	$3 \times 3, 256, (1,1), (1,1)$
conv6_2	$3 \times 3, 256, (1,1), (1,1)$
conv7_1	$3 \times 3, 256, (1,1), (1,1)$
conv7_2	$3 \times 3, 256, (1,1), (1,1)$
conv8_1	$3 \times 3, 512, (1,1), (1,1)$
conv8_2	$3 \times 3, 512, (1,1), (1,1)$
conv9_1	$3 \times 3, 512, (1,1), (1,1)$
conv9_2	$3 \times 3, 512, (1,1), (1,1)$
adaptive average pooling to $(1,1)$	
full connection	2048
mean pooling	
full connection	2048
	128
	7363
softmax	

In addition to the ResNet x-vector, we also trained a DNN for speaker bottleneck extraction. The DNN was trained with the same dataset as the ResNet. The same log filterbank feature as the ResNet was used as the acoustic input feature. In the input layer, each frame was concatenated with 20 frames on both left and right sides. The network structure was $3280\,(80 \times 41) - 1024 - 1024 - 1024 - 1024 - 80 - 7363$. The model was trained to discriminate among the speakers in the training set with corss-entropy loss. The details for the network can be found in [21]. The linear output of the layer before the output layer with dimension 80 was taken as the speaker bottleneck feature vector. The bottleneck feature was used to compute the BIC score matrix for number of speakers estimation and cluster selection as well as training the GMMs in HMM re-segmentation.

As for the score matrix used in number of speakers estimation and cluster selection, we compared the cosine distance matrix with the normalization as used in spectral clustering [25] and the BIC score on speaker bottleneck feature. In x-vector/cosine score matrix computation, we tried two methods: 1. applying the score matrix after AHC which was obtained by score fusion on the distance matrix initialized by the individual segments in every merging step; 2. averaging the x-vectors of the segments in a cluster and computing cosine distance among the averaged x-vectors of the clusters. The results showed that the BIC score matrix provided more accurate speaker number estimation. In this regard, the BIC computed on the covariance matrices of the cluster frames, with the cluster sizes also involved, seemed to provide a score matrix $\mathbf{S}$ of richer information and more robust cluster distance measurement in our number of speakers estimation method.

In our experiments, we compared two methods, i.e., AHC and our proposed diarization system with early-stop clustering. The x-vector combined with cosine distance was used in the clustering phase of AHC and the early-stop mechanism. After the clustering, two iterations of Viterbi re-segmentation were applied to the AHC and early-stop clustering systems respectively.

Firstly, we carried out the experiments under the condition where the number of speakers in each conversation was given. In the AHC system, the clustering stops when the number of clusters equals the ground-truth number of speakers. In our proposed early-stop clustering, the clustering stops at a cosine threshold tuned on the development set. Then the clusters were selected with the number being indicated by the ground-truth number of speakers. Table 2 presents the performance comparison between the two systems. In the early-stop clustering, the stop threshold on the cosine distance was 0.25. The penalty weight λ for BIC [20] score matrix computation was 1.5.

Table 2: Performance comparison between the conventional AHC and the proposed early-stop clustering (ES) on the development (dev) and evaluation (eval) sets respectively, under the condition that the number of speakers was given. Speaker error (%) (SpkErr) and miss rate (%) (Miss) as well as the final DER (%) are included as the metrics for performance evaluation.

	dev		eval	
	AHC	ES	AHC	ES
SpkErr	14.0	9.7	18.8	13.4
Miss	7.9	7.9	9.4	9.4
DER	21.9	17.6	28.2	22.8

From the results, we can see that the proposed diarization system with early-stop clustering achieves better performance than the conventional AHC. Based on the results from the clustering phase, we further evaluated the speaker purity of the resultant clusters. We adopted the cluster purity metric in [18]. In particular, the purity of a cluster $\mathcal{C}$ was defined as:

$$P\left(\mathcal{C}\right) = \frac{||\mathcal{C} \cap \mathcal{C}_{\mathrm{ref}}||}{||\mathcal{C}||} \tag{3}$$

where $\mathcal{C}_{\mathrm{ref}}$ is the cluster provided as the reference. $\mathcal{C} \cap \mathcal{C}_{\mathrm{ref}}$ represents the frames that belong to both clusters $\mathcal{C}$ and $\mathcal{C}_{\mathrm{ref}}$. $||\bullet||$ denotes the number of frames in the cluster. Given a set of conversation files, the denominator and the numerator in (3) were accumulated across all the conversations, respectively. The cluster purity of the two systems are given in Table 3.

From Table 3, we can see that the clusters selected from the early-stop clustering were able to provide purer speaker clusters. As a result, it provides better speaker initialization to the

[1] https://github.com/pytorch/vision

Table 3: Purity (%) comparison between the conventional AHC and the proposed early-stop clustering (ES) on development (dev) set.

	AHC	ES
purity(%)	78.6	84.08

subsequent re-segmentation step, leading to the superior DER performance to AHC.

Next, we evaluated the performance under the condition where the number of speakers is not given. In both AHC and the early-stop clustering, the clustering stops when the cosine distance between two merging clusters is smaller than the threshold. In early-stop clustering, the number of speakers is then estimated and the corresponding clusters are selected. The performance comparison is presented in Table 4. The best performances in AHC and the early-stop clustering were obtained at the cosine thresholds 0.15 and 0.3, respectively. The penalty weight λ for BIC score matrix computation was 1.5.

Table 4: Performance comparison among the conventional AHC and the proposed early-stop clustering (ES) on development (dev) and evaluation (eval) sets respectively, under the condition that the number of speakers was not given. Speaker error (%) (SpkErr) and miss alarm (%) (Miss) as well as the final DER (%) are included as the metrics for performance evaluation.

	dev		eval	
	AHC	ES	AHC	ES
SpkErr	11.9	10.0	16.1	14.8
Miss	7.9	7.9	9.4	9.4
DER	19.8	17.9	25.5	24.2

Comparing the performances in Tables 2 and 4, we can see that, the conventional AHC achieved better performance when the number of speakers was not given. This is due to that when the number of speakers was not given, the best diarization performance for AHC always occurred at the threshold which resulted in more clusters than the actual number of speakers. The threshold needs to take a balance between two factors: 1. the speech segments from the same speaker needs to be clustered as much as possible; 2. the clustering should stop before the clusters from different speakers, which are large enough to affect the diarization performance significantly, being merged. Statistically, Table 5 presents the percentages of speech files in the development set whose estimated numbers of speakers were more, less than and equal to the ground truth. Both the AHC and the number of speaker estimation technique based on the early-stop clustering are included. From the table, we can see that in our AHC experiment, most conversation files were estimated to have more clusters than the actual speaker number. This was also the case in the proposed number of speakers estimation module in our early-stop clustering method, though in a smaller percentage. On the other hand, with higher accuracy in speaker number estimation (estimation equal to the ground truth), the number of speakers estimation module is effective, in comparison with the AHC where the number of speakers is determined by the stop threshold without such a specific module. Still, the limitation in the speaker number estimation in the early-stop clustering seems to be the reason for the performance degradation in Table 4 when compared with that in Table 2.

Table 5: Number of speaker number estimation in conventional AHC and the system based on early-stop clustering (ES) on the development set. Three relations between the estimated numbers of speakers (EST) and the ground truth (GT) are included: EST larger than GT, EST smaller than GT and EST equal to GT.

	AHC	ES
larger	84.1	57.7
equal	9.8	28.2
smaller	6.1	14.1

Moreover, from the results in Table 4, we can see that combined with number of speakers estimation and cluster selection, the proposed early-stop clustering achieves better performance than AHC. As shown before, the effectiveness in cluster purity improvement of our proposed cluster selection technique should be the main reason for this efficacy. Another reason is the better accuracy in estimating the number of speakers as shown in Table 5.

Finally, we experimented on the stability of threshold between AHC and the propose early-stop clustering on the development set. In Fig. 4, the DER curves on different thresholds are shown for both systems. Considering that in practice the approximate upper bound on the number of speakers is known a priori [13], we set the maximum number of clusters during clustering to 20. That is, even the cosine similarity between the merging clusters was below the threshold, the clustering wouldn't stop until the number of clusters was no longer larger than 20.

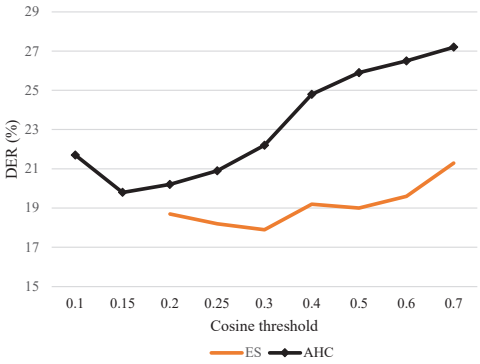

Figure 4: DER comparison between AHC and diarization system with early-stop clustering (ES) with regarding to cosine threshold.

From Fig. 4, we can see that in AHC, the best DER was obtained at the threshold 0.15. The DER degrades gradually with the threshold getting larger. However, from thresholds 0.2 to 0.6, the DER oscillates slightly for the system base on early-stop clustering. In particular, with the threshold varying from 0.15 to 0.6, the relative DER increase from the lowest to the highest DERs for AHC and early-stop clustering were 33.8% and 9.5%, respectively. On the other hand, in the diarization system based on AHC, for the thresholds from 0.15 to 0.6, the standard variance of DER was 2.59. In comparison, in the diarization system based on the proposed early-stop clustering, the standard variance was 0.58 among the cosine thresholds from 0.2 to 0.6. It means that comparing with AHC, the system with early-stop clustering is less sensitive to the threshold. That

is to say that a threshold can be set easily with less fine tuning requirement for the proposed early-stop clustering.

For reference, the results on the First DIHARD data set from the literature are shown in Table 6. From the table, we can see that the systems in our experiments are comparable to those systems. Moreover, the techniques that were used in these work might be helpful to further improve the performance of our system, including the signal processing methods (denoising, dereverberation), ResNet training methods, the scoring on x-vectors, the VB HMM training in re-segmentation, etc. Investigating into the benefits of such techniques in our system can be our future research points.

Table 6: DER (%) on First DIHARD Track 1. (+VB) denotes using variational Bayesian in HMM re-segmentation. (+cosine) denotes using cosine in scoring.

system	DER	
	dev	eval
ZCU-NTIS[26]	-	26.9
USTC PLDA (+ cosine) [27]	19.51(17.40)	-(24.56)
ViVoLAB[28]	20.14	26.02
JHU (+VB) [19]	20.03 (18.20)	25.94 (23.73)
EURECOM [29]	25.56	29.33

5. Conclusion

We proposed an early-stop clustering strategy for improving speaker diarization performance. The strategy can reduce inhomogeneous merge of speech frames from interfering speakers into a cluster. The proposed strategy consists of two steps: 1. setting the number of initial clusters larger than the anticipated maximum number of speakers; 2. combining extraneous clusters into the targeted number of speakers. The approach leads to better matched clusters with the corresponding speakers. Tested on the First DIHARD database, the proposed strategy yields a better DER performance when the number of speakers is either given and not, with relative improvements of DER of 19.15% and 5.10% on the evaluation set, respectively. Moreover, the proposed similarity matrix-based estimate of the number of speakers and the resultant speaker clusters can make the threshold setting process relatively simple and robust.

6. References

[1] X. Anguera, S. Bozonnet, N. Evans, C. Fredouille, G. Friedland, and O. Vinyals, "Speaker diarization: a review of recent research," *IEEE Trans. on Audio, Speech, and Language Processing*, vol. 20, no. 2, pp. 356—-370, 2012.

[2] G. Sell, D. Snyder, A. McCree, D. Garcia-Romero, J. Villalba, M. Maciejewski, V. Manohar, N. Dehak, D. Povey, S. Watanabe, et al., "Diarization is hard: Some experiences and lessons learned for the jhu team in the inaugural dihard challenge.," in *Interspeech*, 2018, pp. 2808–2812.

[3] N. Ryant, K. Church, C. Cieri, A. Cristia, J. Du, S. Ganapathy, and M. Liberman, "First DIHARD challenge evaluation plan," *2018, tech. Rep.*, 2018.

[4] N. Ryant, K. Church, C. Cieri, A. Cristia, J. Du, S. Ganapathy, and M. Liberman, "Second DIHARD challenge evaluation plan," *Linguistic Data Consortium, Tech. Rep*, 2019.

[5] "NIST 2018 Speaker Recognition Evaluation Plan," `https://www.nist.gov/system/files/documents/2018/08/17/sre18_eval_plan_2018-05-31_v6.pdf`.

[6] "NIST 2019 Speaker Recognition Evaluation Plan," `https://www.nist.gov/system/files/documents/2019/08/16/2019_nist_multimedia_speaker_recognition_evaluation_plan_v3.pdf`.

[7] M. A. Siegler, U. Jain, B. Raj, and R. M. Stern, "Automatic segmentation, classification and clustering of broadcast news audio," in *Proc. DARPA speech recognition workshop*, 1997.

[8] G. Sell and D. Garcia-Romero, "Speaker diarization with PLDA i-vector scoring and unsupervised calibration," in *Spoken Language Technology Workshop (SLT)*, 2014, pp. 413–417.

[9] D. Garcia-Romero, D. Snyder, G. Sell, D. Povey, and A. McCree, "Speaker diarization using deep neural network embeddings," in *ICASSP*, 2017, pp. 4930–4934.

[10] M. Diez, L. Burget, S. Wang, J. Rohdin, and J. Černocký, "Bayesian HMM Based x-Vector Clustering for Speaker Diarization," in *Proc. Interspeech 2019*, 2019, pp. 346–350.

[11] C. Bishop, *Pattern recognition and machine learning*, Springer, New York, 2006.

[12] M. Diez, L. Burget, and P. Matejka, "Speaker diarization based on bayesian HMM with eigenvoice priors," in *Proc. Odyssey 2018 The Speaker and Language Recognition Workshop*, 2018, pp. 147–154.

[13] P. Kenny, "Bayesian analysis of speaker diarization with eigenvoice priors," 2008.

[14] A. OT Hogg, C. Evers, and P. A Naylor, "Speaker change detection using fundamental frequency with application to multi-talker segmentation," in *ICASSP*, 2019, pp. 5826–5830.

[15] S. Cheng, H. Wang, and H. Fu, "BIC-Based Speaker Segmentation Using Divide-and-Conquer Strategies With Application to Speaker Diarization," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 18, no. 1, pp. 141–157, 2010.

[16] X. Anguera, C. Wooters, and J. Hernando, "Purity algorithms for speaker diarization of meetings data," in *Proc. ICASSP*, 2006, pp. I1025–I1028.

[17] T. L. Nwe, H. Sun, B. Ma, and H. Li, "Speaker clustering and cluster purification methods for rt07 and rt09 evaluation meeting data," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 2, pp. 461–473, 2012.

[18] Z. Xiang, "A cluster purification algorithm for speaker diarization system," in *2014 Seventh International Symposium on Computational Intelligence and Design*. IEEE, 2014, vol. 2, pp. 538–541.

[19] G. Sell, D. Snyder, A. McCree, D. Garcia-Romero, J. Villalba, et al., "Diarization is hard: some experiences and lessons learned for the JHU Team in the Inaugural DIHARD Challenge.," in *Proc. Interspeech*, 2018, pp. 2808–2812.

[20] S. Chen and P. Gopalakrishnan, "Speaker, environment and channel change detection and clustering via the bayesian information criterion," in *Proc. DARPA broadcast news transcription and understanding workshop*, 1998, vol. 8, pp. 127–132.

[21] L. Chen, Y. Zhao, S.-X. Zhang, J. Li, G. Ye, and F. Soong, "Exploring sequential characteristics in speaker bottleneck feature for text-dependent speaker verification," in *ICASSP*, 2018, pp. 5364–5368.

[22] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[23] A. Nagrani, J. S. Chung, and A. Zisserman, "VoxCeleb: a large-scale speaker identification dataset," in *Proc. Interspeech*, 2017, pp. 2616 —- 2620.

[24] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," in *INTERSPEECH*, 2018.

[25] H. Ning, M. Liu, H. Tang, and T. S. Huang, "A spectral clustering approach to speaker diarization," in *Ninth International Conference on Spoken Language Processing*, 2006.

[26] Z. Zajıc, M. Kunešová, J. Zelinka, and M. Hrúz, "Zcu-ntis speaker diarization system for the dihard 2018 challenge," in *Proc. INTERSPEECH*, 2018, pp. 2788–2792.

[27] L. Sun, J. Du, C. Jiang, X. Zhang, S. He, B. Yin, and C.-H. Lee, "Speaker diarization with enhancing speech for the First DIHARD Challenge.," in *Interspeech*, 2018, pp. 2793–2797.

[28] I. Viñals, P. Gimeno, A. Ortega, A. Miguel, and E. Lleida, "Estimation of the number of speakers with variational Bayesian PLDA in the DIHARD Diarization Challenge.," in *Proc. Interspeech*, 2018, pp. 2803–2807.

[29] J. Patino, H. Delgado, and N. Evans, "The EURECOM submission to the First DIHARD Challenge.," in *Proc. Interspeech*, 2018, pp. 2813–2817.