INTERSPEECH 2009 BRIGHTON

Improving Speaker Segmentation via Speaker Identification and Text Segmentation

Runxin Li[1], *Tanja Schultz*[1 2], *Qin Jin*[1]

[1]InterACT, Language Technologies Institute, Carnegie Mellon University, Pittsburgh, USA
[2]Fakultät für Informatik, Universität Karlsruhe (TH), Germany
{lirx, tanja, qjin}@cs.cmu.edu

10.21437/Interspeech.2009-272

Abstract

Speaker segmentation is an essential part of a speaker diarization system. Common segmentation systems usually miss speaker change points when speakers switch fast. These errors seriously confuse the following speaker clustering step and result in high overall speaker diarization error rates. In this paper two methods are proposed to deal with this problem: The first approach uses speaker identification techniques to boost speaker segmentation. And the second approach applies text segmentation methods to improve the performance of speaker segmentation. Experiments on Quaero speaker diarization evaluation data shows that our methods achieve up to 45% relative reduction in the speaker diarization error and 64% relative increase in the speaker change detection recall rate over the baseline system. Moreover, both these two approaches can be considered as post-processing steps over the baseline segmentation, therefore, they can be applied in any speaker diarization systems.

Index Terms: speaker diarization, speaker segmentation, speaker identification, text segmentation

1. Introduction

Given an audio stream with multiple speakers involved, the goal of the speaker diarization system is to split the audio into homogeneous segments and to answer the question of "Who spoke when?" A typical speaker diarization (a.k.a. speaker segmentation and clustering) system generally contains two components: the first component is "speaker segmentation" whose goal is to split the audio into homogeneous segments and the key challenge is to detect the locations of the speaker changes or turns; the second component is "speaker clustering" which aims at grouping all the segments that belong to the same speaker together.

Good speaker segmentation should provide the correct speaker changes as the result; each segment should contain exactly one speaker. There are two types of errors related to speaker change detection: insertion error (when a speaker change is detected but it does not exist in reference) and deletion error (an existing speaker change is not detected). These two types of errors have different impact depending upon the application. In our system, the segmentation stage is followed by a clustering stage. Therefore, insertion errors (resulting in an over-segmentation) are less critical than deletion errors, since the clustering procedure has the opportunity to correct the insertion errors by grouping the segments related to the same speaker. While deletion errors cannot be recovered in the clustering stage.

A lot of research has been done to minimize the segmentation errors. Some of them used joint segmentation and clustering schemes, including iterative Viberbi decoding during agglomerative clustering[1], and ergodic-HMM[2] with a top-down strategy, in which each speaker is represented by a state and the changes between speakers are represented by transitions in the HMM. Others applied iterative GMM segmentation/clustering and re-segmentation after the initial segmentation[3].

In this paper, two simple but efficient approaches are proposed to deal with the problem of speaker segmentation and improve the overall speaker diarization performance. Both methods are applied after the baseline speaker segmentation so they can be considered as postprocessing steps.

The first approach uses speaker identification techniques to refine the former segmentation criterions and label each speech segment with its speaker ID. Speaker identification techniques are proved to be helpful to the speaker clustering[4]. In this paper the speaker identification models are trained by MAP algorithm over the background model, more discriminative powers are preserved to help boost the speaker segmentation and then the following clustering task.

The idea of the second approach comes from the task of text segmentation, which aims at partitioning a document into a set of segments, each of which is coherent about a specific topic. To use this method, it is necessary to find a way to transform the feature vector of each frame of the speech signal into textual tokens. In Gaussian Mixture Models it is revealed that only the most probable mixture components have significant impacts on the tasks of speaker recognition. In this paper a GMM is first trained on the whole speech, then these components are considered as the tokens for speech frames, at last text segmentation methods are proposed after the tokenization to find the best segment boundaries during the speech.

The remainder of this paper is organized as follows: section 2 describes our baseline speaker diarization system and the improved systems with the two proposed approaches. Section 3 presents the experimental results and section 4 presents our conclusions.

2. System Description

2.1. Baseline System

As shown in the circled area of Figure1, our baseline system consists of three main components: Audio segmentation is realized by an HMM segmenter with four classes: Speech, Noise, Silence, and Music. The speech features used are 13-dimension MFCCs plus their first and second derivatives. Each class is represented by a GMM with 64 Gaussians. The system is trained on 3 hours of manually annotated HUB4 English shows.

Copyright © 2009 ISCA

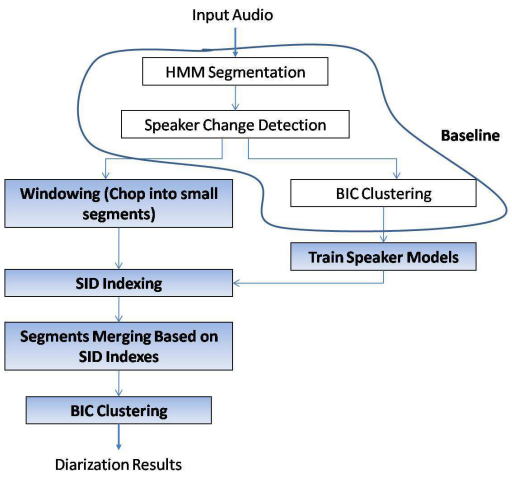


Figure 1: The system flow chart with SID-Seg

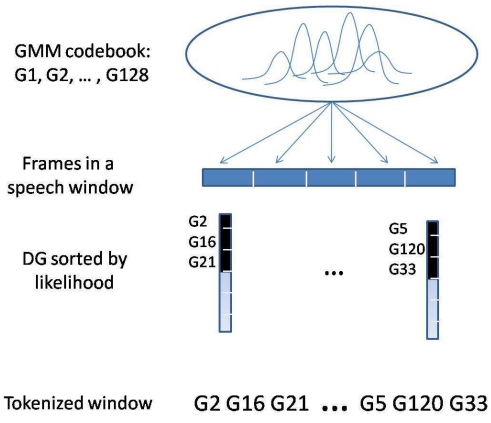


Figure 2: Example of extracting top 3 DG for a speech window

Speaker change detection is applied on any segment that is longer than 5 seconds to check whether there exist speaker turn changes that have not been detected[5]. For speaker clustering, we use a hierarchical, agglomerative clustering technique based on 128 Tied GMM and Bayesian Information Criterion (BIC) stopping criterion[6].

2.2. Speaker Identification for Speaker Segmentation (SID-Seg)

Although current diarization systems are only evaluated using "relative" speaker labels (such as "spkr1"), using speaker identification (SID) techniques can still be helpful for the speaker diarization tasks. This paper utilizes the speaker identification to help improve the accuracy of speaker segmentation. As described in figure1, our SID-Seg approach consists of several steps:

1. **Train Speaker Models:** Train speaker models using the speaker labels of the baseline clustering results. The Universal Background Model (UBM)[7] is trained on the whole test speech recordings instead of using other corpus. Speaker models are then trained by Maximum a Posteriori (MAP)[8] on the UBM.
2. **Windowing:** Chop the segments from the baseline system into small speech pieces with a fixed window size, e.g. 1.5 seconds in this paper.
3. **SID Indexing:** After the speaker models training process, each speech window from step 2 is classified by the speaker models and labeled as its corresponding speaker identity.
4. **Segments Merging:** Adjacent speech windows labeled as the same speaker are concatenated to generate the final speaker segmentation outputs.
5. **BIC Clustering:** The second pass of speaker clustering is then applied to achieve the final diarization result.

2.3. Text Segmentation for Speaker Segmentation (TS-Seg)

The goal of text segmentation is to partition a document into a set of segments, each of which is coherent about a specific topic. If we consider speaker as the topic and find a way to transform the speech signal into text, we can apply the text segmentation methods for speaker segmentation. This paper proposes a TS-Seg system, which consists of several steps: 1) **windowing** as what we did for SID-Seg, 2) **training of a Tied GMM** on the whole speech signal to generate a GMM codebook, 3) **tokenization** for each frame using the codebook to generate textual documents for each speech window from step2, 4) **text segmentation** and 5) **BIC clustering** on the text segmentation result.

We make use of GMM to tokenize the speech signal. Theoretically, during likelihood computation with GMMs, all Gaussian mixture components are used. A lot of research indicate that only a small portion of the mixtures - components that have the largest likelihoods on current frame feature - contribute significantly to the likelihood computation[9] in speaker recognition tasks. In HMM based speech recognition systems, the sequences of "dominant Gaussian components" (DG), which are also named "speech trajectories", are proved to be strongly correlated with speaker variations[10][11]. All these research indicate that we could investigate the discriminative powers of the dominant Gaussian components in our task. The tokenization procedure in TS-Seg system is described by the example in figure2. The top level of this figure is the 128 Tied GMM whose codebook is represented by its mixture components "G1, G2, $\cdots$, G128". The second level is a speech window generated by the windowing process and consists of 5 frames. For each frame we calculate the likelihoods of all the Gaussian components and sort them by their likelihood values, sorted lists are shown in the third level of this figure. Finally the top N (3 in this example) components are extracted from the list and the current frame is labeled as the corresponding codewords in the GMM codebook. The whole speech window is tokenized as the concatenation of the labels for all the frames within it.

In TS-Seg, similarities are calculated between two adjacent segments, each of which is a concatenation of several neighboring speech windows. In this paper, a very simple similarity metric, the cosine of the angles between the word distribution vectors of two segments, is employed. Formally saying, assume we have two segments b_1 and b_2, then the similarity:

$$Sim(b_1; b_2) = \frac{\sum_{i=1}^{V} \widehat{P}(w_i|b_1)\widehat{P}(w_i|b_2)}{\sqrt{\sum_{i=1}^{V} \widehat{P}(w_i|b_1)^2} \cdot \sqrt{\sum_{i=1}^{V} \widehat{P}(w_i|b_2)^2}} \quad (1)$$

where the empirical distribution of words in the segment $\widehat{P}(w_i|b)$ can be obtained from the number of word-segment co-occurrence $n(b, w_i)$, normalized by the number of frames in the segment, and V represents the size of the vocabulary. In equation1 the larger value means more similarity between two segments. In addition, we treat this task as an optimization problem and use dynamic programming to find the segmentation that has the least overall similarities among them[12]:

$$\begin{aligned} C(s_i) &= \min_{s_i-N+1\le t\le s_i}\Big\{C(t)+Sim(b_{p(t),t-1};b_{t,s_i})\Big\} \\ p(s_i) &= \arg\min_{s_i-N+1\le t\le s_i}\Big\{C(t)+Sim(b_{p(t),t-1};b_{t,s_i})\Big\} \end{aligned} \quad (2)$$

where $C(s_i)$ represents the smallest total similarity value from the beginning of the speech to the current speech window s_i, $p(s_i)$ represents the starting window of the optimal segment that ends in current window s_i, $b_{t,s}$ represents the segment from window t to window s and N represents the limit of the maximum number of speech windows a segment has. The optimal segmentation of the speech signal is achieved after we go over all the speech windows and then backtrack from the last one.

3. Experiments

3.1. Corpora and Experimental Design

To evaluate the proposed methods, we used the data that have been used for evaluation of the speaker diarization systems within the Quaero project (ESTER)[1]. The data includes French data and English data. The French data is from ESTER corpus and the English data is from Naked Scientist shows[13]. There are 20 shows of more than 6 hours of speech in all in French and the types of shows vary from news TV shows to interviews. The English data consists of 3 hours of TV shows.

In this paper a series of experiments are designed to test the effectiveness of the two proposed methods. The first experiment aims at evaluating the performance of the SID system. In the second experiment, the TS-Seg is performed on the same data, using several different parameter settings. At last the two methods are combined together to see further improvements.

Standard speaker diarization error rate (DER) is used in all experiments in this paper as the evaluation metric for the overall speaker segmentation and clustering performance. It can be expressed in terms of the miss (speaker in reference but not in system hypothesis), false alarm (speaker in system hypothesis but not in reference), and speaker error (mapped reference speaker is not the same as the hypothesized speaker) rates. DER is the sum of these three components based on the optimal speaker mapping of hypothesized speakers and reference speakers.

In order to better analyze the performance of speaker segmentation methods, a speaker change detection rate is defined. There are two types of errors related to speaker change detection: insertion error (a speaker change is detected but it does not exist in reference) and deletion error (an existing speaker change is not detected). We define an accuracy window around the reference speaker change point, say 0.5 second in our experiments, then a "hit" is met when the hypothesized change points lies in the window of a reference change point. In this way we can determine the precision (percentage of hit among all the hypothesized change points) and recall (percentage of hit among all the reference change points). Deletion errors will directly lower the recall. Insertion errors will reduce the precision. As mentioned in the paper before, deletion errors are more critical to the performance of speaker segmentation, we only investigate the recall rate of different methods in this paper.

[1]The ESTER data were provided by DGA for the purpose of evaluation within the Quaero project, which is funded by OSEO, French State agency for innovation.

Table 1: Speaker diarization errors for the system with SID-Seg

	French	English
HMM	36.29%	19.21%
HMM + Windows	41.01%	27.42%
HMM + Windows + SID	25.32%	12.71%
HMM + Windows + SID + merge	**21.33%**	**11.77%**

Table 2: Speaker diarization errors for the system with TS-Seg under different parameter settings

		French	English
Baseline (HMM)		36.29%	19.21%
max windows 5	number of DG 1	**23.68%**	**11.82%**
	number of DG 5	25.42%	—
max windows 10	number of DG 1	25.09%	15.46%
	number of DG 5	25.38%	—

3.2. Experimental Results

The performance of the baseline system and the system with SID-Seg are shown in Table1. We can see the decrease in DER step by step. Each segmentation is followed by a speaker clustering except for the 4th row because SID labeling results can be considered as a clustering result. From table we can see that the performance of speaker clustering directly from Windowing decreases as we expected. With the SID-Seg, the overall DER reduces significantly, which supports our hypothesis that SID techniques will help the task of speaker diarization.

In the second experiment, different parameter settings are tested, including the number of dominant Gaussian components extracted and the maximum number of windows of a segment. Both parameters are changed to see the robustness of this method. The experimental results are shown in table2. From the table we can see that the performances for this method have no significant difference under different settings on the French data set, although the setting of maximum window number 5 and top 1 dominant Gaussian components performs the best. On the English data set, we only considered different maximum window numbers. There are greater variances on the English side, as the result of maximum window number of 5 obviously exceeds 10. This may be because in the English data set the speakers change more frequently, which makes the average segment length shorter. However the trends of the DERs appear the same on the two different data sets, and this tells that tighter constriction in TS-Seg preserves more possible segment boundaries. On the other hand, it also shows that preserving more DG doesn't necessarily bring more improvements to our TS-Seg system.

The per-show performances of two proposed methods on French data are compared in figure3, where we adopt the parameter setting that achieved the best performance in our previous experiments. There are large variabilities in performances of both systems over the shows. The same observation appears

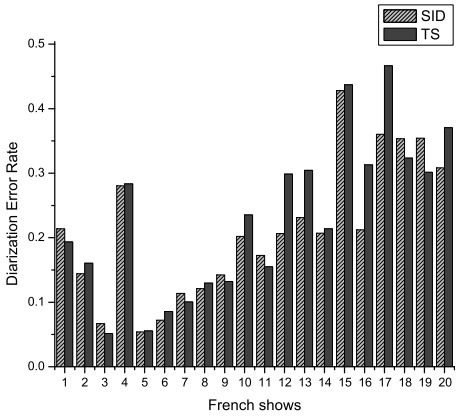


Figure 3: DER per show for two proposed methods on French data set, either one has advantage on parts of the shows

Table 3: DER on all systems, the last row represents the combined system, the 3rd and 5th columns represent the improvements over baseline system on French and English data

	French	*Fr Imprv.*	English	*En Imprv.*
Baseline	36.29%	—	19.21%	—
+ SID	21.33%	*41.22%*	11.77%	*38.73%*
+ TS	23.68%	*34.75%*	11.82%	*38.47%*
+ SID + TS	**21.03%**	*42.05%*	**10.61%**	*44.77%*

in the English data too. So the next experiment is carried out to see if SID-Seg and TS-Seg can be combined to further improve speaker diarization results. In this paper the combination is done in a simple way by just grouping all the boundaries from the two techniques, in other words, the union of the two boundary sets outputted from the two systems is seen as the final segmentation result. No pruning needs to be done because the minimal distance between two boundaries is 1.5 seconds, as we did in the windowing process. Table3 compares the DERs of the baseline system and the improved systems by adding SID-Seg, TS-Seg, and combination of SID-Seg and TS-Seg. We can see from the table that the combination of the two systems gave us additional gains over each one. Up to 45% relative reduction in DER was achieved over the baseline system. Speaker change detection performance is also evaluated for the methods, as shown in table4. We can see that the speaker change detection recall rate was increased by both proposed methods and 64% relative improvement was achieved from combination of the two. The improvements in change detection recall rates are well correlated to their corresponding reduction in the DERs. In the future work, we will explore using more knowledge and heuristics in the combination process to further improve the system performance.

4. Conclusions

In this paper we proposed two new methods to improve speaker segmentation. speaker identification method and text segmentation method are used to deal with the problem of miss detection errors in our baseline segmentation system. Experiments on Quaero speaker diarization evaluation data show that our proposed methods achieved significant improvements over the baseline system with up to 45% relative reduction in speaker diarization error and 64% relative increase in speaker change detection recall rate. Moreover, these two methods can be considered as post-processing methods, therefore they can be easily applied in any speaker diarization systems.

Table 4: The speaker change detection recall rates

	French	*Fr Imprv.*	English	*En Imprv.*
Baseline	33.79%	—	52.94%	—
+ SID	52.40%	*55.08%*	66.91%	*26.39%*
+ TS	45.71%	*35.28%*	52.35%	*-1.11%*
+ SID + TS	**55.31%**	*63.69%*	**70.00%**	*32.23%*

5. References

[1] Gauvain, J. -L., Lamel, L. and Adda, G. “Partitioning and transcription of broadcast news data,”, in Proc. Int. Conf. Spoken Lang. Process., vol. 4, Sydney, Australia, Dec. 1998, pp. 1335-1338.

[2] Meignier,S., Bonastre, J.-F., Fredouille, C., and Merlin, T., “Evolutive HMM for multispeaker tracking system,” in Proc. IEEE Int. Conf. Acoust., Speech, Signal Process., vol. II, Istanbul, Turkey, Jun. 2000, pp. 1201-1204.

[3] Barras,C., Zhu, X. , Meignier, S. , and Gauvain, J.-L.,“Improving speaker diarization, ”, in Proc. Fall Rich Transcription Workshop (RT-04), Palisades, NY, Nov. 2004.

[4] Zhu, X. , Barras, C., Meignier, S. , and Gauvain, J.-L., “Combining speaker identification and BIC for speaker diarization,” in Proc. Eur. Conf. Speech Commun. Technol., Lisbon, Portugal, Sep. 2005, pp. 2441-2444.

[5] Jin, Q. and Schultz, T., “Speaker Segmentation and Clustering in Meetings,” in ICSLP, 2004.

[6] Chen, S. S., and Gopalakrishnam, P. S., “Speaker, environment and channel change detection and clustering via the bayesian information criterion,” in Proc. 1998 DARPA Broadcast News Transcription and Understanding Workshop, Lansdowne, VA, 1998, pp. 127-132.

[7] D. Reynolds, and R. Rose, “Robust Text-independent Speaker Identification Using Gaussian Mixture Speaker Models,” IEEE Trans. on Speech and Audio Processing, vol.3, pp.72-83, 1995.

[8] D. Reynolds, T. Quatieri, and R. Dunn, “Speaker Verification using adapted Gaussian Mixture Models,” Digital Signal Processing, vol. 10, no. 1-3, pp. 19-41, 2000.

[9] Bimbot, F., Bonastre, J. F. , Fredouille, C. , Gravier, G. , Chagnolleau, Magrin I. , Meignier, S. , Merlin, T. , Ortega-Garcia, J. , Petrovska-Delacretaz, D., and Reynolds,D.A., “A Tutorial on Text-Independent Speaker Verification,” in EURASIP Journal on Applied Signal Processing, volumn 4, pp. 430-451, 2004.

[10] Han, Y., Hämäläinen, A., Boves, L., “Trajectory clustering of syllable-length acoustic models for continuous speech recognition,” In Proc. of ICASSP-2006, vol. I, pp. 1169-1170, 2006.

[11] Gish, H. and Ng, K. “Parametric trajectory models for speech recognition,” in Fourth International Conference on Spoken Language, ICSLP, pp. 466-469, 1996.

[12] Fragkou,P., Petridis,V., and Kehagias, A., “A Dynamic Programming Algorithm for Linear Text Segmentation,” in J. Intell. Inf. Syst., 23(2): 179-197, 2004

[13] “The naked scientists online,” http://www.thenakedscientists.com