HYBRID SPEAKER-BASED SEGMENTATION SYSTEM USING MODEL-LEVEL CLUSTERING

Hyoung-Gook Kim, Daniel Ertelt, Thomas Sikora

Communication Systems Institute
Technical University of Berlin, Germany
{kim, ertelt, sikora}@nue.tu-berlin.de

ABSTRACT

In this paper, we present a hybrid speaker-based segmentation, which combines metric-based and model-based techniques. Without a priori information about number of speakers and speaker identities, the speech stream is segmented by three stages: (1) The most likely speaker changes are detected. (2) To group segments of identical speakers, a two-level clustering algorithm using a Bayesian Information Criterion (BIC) and HMM model scores is performed. Every cluster is assumed to contain only one speaker. (3) The speaker models are reestimated from each cluster by HMM. Finally a resegmentation step performs a more refined segmentation using these speaker models. For measuring the performance we compare the segmentation results of the proposed hybrid method versus metric-based segmentation. Results show that the hybrid approach using two-level clustering significantly outperforms direct metric based segmentation.

1. INTRODUCTION

Our challenge is the design of automatic speaker-based segmentation algorithms integrated with speech recognition and speaker identification to enable indexing, quick browsing, and searching of audio documents.

Segmenting audio data into speaker-labeled segments is the process of determining where speakers are engaged in a conversation (start and end of their turn). There are three major categories of speaker-based segmentation: metric-based, model-based, and hybrid (combined metric-based and model-based) segmentation.

In model-based segmentation [1], a set of models for different acoustic speaker classes from a training corpus is defined and trained prior to segmentation. The incoming speech stream is classified using the models. However, most model-based approaches require a priori information to initialize the speaker models.

The metric-based segmentation task [2][3] is divided in two main parts: speaker change detection and segment clustering. First, the speaker change detection step splits the conversation into smaller segments that are assumed to contain only one speaker. The next step is to merge speech segments related to a same speaker. The metric-based segmentation relies on thresholding, which lacks stability and robustness.

In [4][5], it is shown that the hybrid algorithm, which combines metric-based and model-based techniques, works significantly better than all other approaches. A metric-based segmentation is only used to create an initial set of speaker models. Next, model-based resegmentation performs a more refined segmentation using these speaker models.

Generally, the speaker models are estimated from each cluster. However, if the speaker number detected by metric-based segmentation is larger than the actual speaker number, the model-based segmentation can not achieve higher accuracy with these speaker models.

In this paper, we focus on combination of metric-based and model-based segmentation. To group segments of the same speaker, a two-level clustering algorithm consisting of segment-level and model-level clustering is performed. This paper is organized as follows. Section 2 introduces the system framework and the individual components of our method. Section 3 reports experimental results. Section 4 gives the conclusion.

2. SYSTEM FRAMEWORK

Our hybrid segmentation is a combination of metric-based and model-based segmentation. Figure 1 depicts the algorithm flow chart. The hybrid segmentation can be divided into seven modules: silence removal, feature extraction, speaker change detection, segment-level clustering, speaker model training, model-level clustering and HMM-based resegmentation using the retrained speaker models.

First, silence segments in the input audio recording are detected by the simple energy-based algorithm. The detected silence part is used to train a silence model. The speech part is transformed into a feature vector sequence and fed into the speaker change detection step, which

splits the conversational speech stream into smaller segments. The speech segments detected by speaker change detection are classified into clusters by segment-level clustering such that each cluster is assumed to contain the speech of only one speaker.

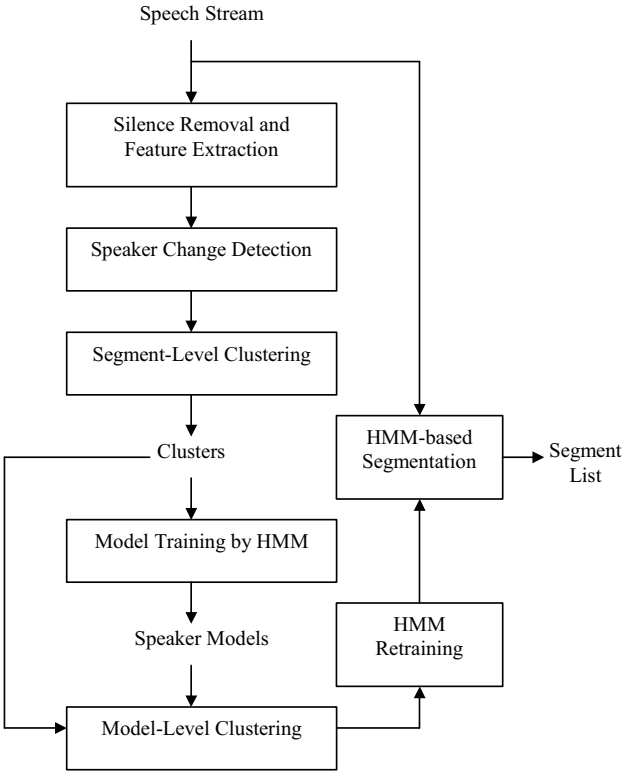

Figure 1: Block diagram of the hybrid speaker-based segmentation system

After training a model for every cluster, model-cluster merging is performed by *L*-best likelihood scores from all cluster speaker models, thus yielding a target cluster number equal to the actual speaker number. So model-level clustering achieve higher accuracy than segment-level clustering. After merging the two clusters, the cluster models are retrained. The retrained speaker models are used to resegment the speech stream. Finally HMM-based resegmentation step is achieved by Viterbi algorithm to determine the maximum likelihood score.

2.1. Speaker Change Detection

Speech signals are first parameterized in terms of acoustic feature vectors and then the distance between two neighboring segments is sequentially calculated for speaker change detection.

Various speaker change detection algorithms differ in the kind of dissimilarity function they employ, the size of the two windows, the time increments of the shifting of the two windows, and the way the resulting dissimilarity values are evaluated and thresholded.

For measuring the performance of the speaker change detection we compare two methods: divergence shape distance (DSD) [6] and Bayesian Information Criterion (BIC) [7]. Prior to speaker change detection, a silence detection module detects silence segments in the input speech stream and the detected silence segments are used to train a silence model.

• Divergence shape distance (DSD)

The non-silence speech stream is divided into 3s sub-segments with 2.5s overlapping. The sub-segment is further divided into overlapping frames with 50% overlapping for consecutive frames, where 23-order Mel-scale Frequency Cepstrum Coefficients (MFCC) are extracted. For the detection of speaker changes, two neighbouring sub-segments of the MFCC feature vectors are moved over the speech stream. The similarity between the contents of the two sub-segments is computed using a divergence shape distance function. The dissimilarity *D* between two neighboring sub-segments is defined as the distance determined by the covariance of two sub-segments, which is defined by

$$D = \frac{1}{2} tr\left[\left(\Sigma_f - \Sigma_l\right)\left(\Sigma_f^{-1} - \Sigma_l^{-1}\right)\right] \quad (1)$$

where Σ_f and Σ_l represent the covariance of reliable speaker-related vectors in the former sub-segment *f* and the latter sub-segment *l* respectively.

• Bayesian Information Criterion (BIC) approach: BIC approach has been the subject of considerable attention in recent years due to its effectiveness for speaker change detection. Within a detection window, T^2 statistics are calculated at every point. The peak value point is chosen as a candidate of a speaker change. If the speaker change is confirmed by the BIC check, a new detection window is started from this point to search for the next speaker change point. Otherwise, the detection window is expanded to enlarge the search range.

BIC is supposed to have the advantage of not having any thresholding. However, the choice of a penalty factor determines the sensitivity of the method to changes.

2.2. Segment-Level Clustering

The goal of speaker clustering is to identify and group together all speech segments that were produced by the same speaker. In our case, clustering of segments of the same speaker is done by using the BIC as a distance between two clusters.

Given a set of speech segments $S_1 \ldots S_k$ detected by speaker change detection, one step of the algorithm

consists in merging two of them. In order to decide if it is convenient to merge S_i and S_j, the difference between the BIC values of two clusters is computed:

$$\Delta BIC = n\log|\Sigma| - n_i \log|\Sigma_i| - n_j \log|\Sigma_j| - \lambda P, \qquad (2)$$
$$P = \frac{1}{2}\left\{ d + \frac{1}{2} d(d+1) \right\} \log n$$

where Σ is the covariance matrix estimated on acoustic data of S_i and S_j, Σ_i on S_i and Σ_j on S_j; $n = n_i + n_j$, λ is a penalty factor and P being the dimension of the acoustic space.

The more negative the *ΔBIC* is, the closer the two clusters are. At the beginning, each segment is considered to be a single segment cluster and distances between it and the other clusters are computed. The two closest clusters are then merged if the corresponding *ΔBIC* is negative. In this case, distances between the new cluster and the other ones are computed, and the new pair of closest clusters is selected at the new iteration. Otherwise, the algorithm ends.

2.3. Model-Level Clustering

After segment-level clustering the cluster number may be larger than the actual speaker number. Starting with speaker models trained from these clusters model-based segmentation can not achieve higher accuracy. In order to obtain the correct cluster number equal to the actual speaker number we perform model-level clustering using speaker model scores (likelihood).

In order to train a statistical speaker model for each cluster, we use Hidden Markov Models (HMMs), which consist of several states. In the speech stream of television broadcasts or panel discussion TV programs, temporal structures of video sequences require the use of an ergodic topology, where each state can be reached from any other state and can be revisited after leaving. Given the feature vectors of one cluster, an ergodic HMM with 7 states for the cluster is trained using a maximum likelihood estimation procedure known as the Baum-Welch algorithm. All cluster HMM models are combined into a larger network which is used to merge the two clusters.

The feature vectors of each cluster are fed to the HMM network containing all reference cluster speaker models in parallel. The reference speaker model scores (likelihoods) are calculated over the whole set of feature vectors of each cluster.

All these likelihoods are passed in the *Likelihood Selection block*, where the similarity between all combinations of two reference scores is measured by the likelihood-distance:

$$d_i(i,j) = P(C_i \mid \lambda_i) - P(C_i \mid \lambda_j) \qquad (3)$$

where C_l denotes the observations belonging to cluster *l*, $P(C_l \mid \lambda_l)$ the cluster model score, λ_l the speaker model. If $d_i(i,j) \le \Delta$, the index *j* is stored as the candidate in the *L*-best likelihood table T_i. This table provides also ranking of the cluster models similar to C_i. In order to decide if the candidate models *j* in the table T_i belong to the same speaker, we check the *L*-best likelihood table T_j, where distances between *j* cluster model and other reference model *i* are computed:

$$d_j(j,i) = P(C_j \mid \lambda_j) - P(C_i \mid \lambda_i) \qquad (4)$$

If $d_j(j,i) \le \Delta$, we assume that HMM λ_i and HMM λ_j represent the same speaker and thus we merge cluster C_i and cluster C_j, else λ_i and λ_j represent different speakers. This way we check all entries in table T_i and similar clusters are merged. So model-level clustering achieves higher accuracy than direct segment-level clustering. After merging the clusters, the cluster models are retrained.

2.4. HMM-Based Resegmentation

For HMM-based resegmentation, the speech stream is divided into 1.5 second sub-segments, which overlap by 33%. We assume that there is no speaker change within each sub-segment. Therefore, speaker segmentation can be performed at the sub-segment level. Given a 1.5 second long sub-segment as input, the MFCC features are extracted and fed to all reference speaker models in parallel. Then, the Viterbi algorithm finds the maximum likelihood sequence of states through the HMM-based recognition classifier and returns the most likely classification label for the sub-segment. Invalid input, such as heated discussions with multiple people speaking at the same time, cause sub-segments to sometimes be classified incorrectly when there are no appropriate models for the input. As a result, the sub-segment labels needed to be smoothed out. To this end, we use a low-pass filter to enable more robust segmentation by correcting errors. The filter waits for *A* adjacent sub-segments of the same label before declaring the beginning of a segment. Errors can be tolerated within a segment, but once *B* adjacent classifications of any other models are found, the segment is ended. For our data, the optimum values were *A = 3* and *B = 3*.

3. EXPERIMENTS

3.1. Data set

To evaluate the performance of the different segmentation approaches, we used one audio track from television talk-show program. It is approximately 90 minutes long and contains 6 speakers (4 male and 2 female). The speakers interrupt each other frequently.

3.2. Segmentation Results

For the measure of the performance we distinguish four types of errors: recognition rate (RR), recall (*RCL*), precision (*PRC*), related to speaker-based segmentation.

The F-measure *F* is a combination of the recall (*RCL*) rate of correct boundaries and the precision (*PRC*) rate of the detected boundaries. When *RCL* and *PRC* errors are weighted as being equally detractive to the quality of segmentation, *F* is defined as

$$F = \frac{2 \cdot PRC \cdot RCL}{PRC + RCL} \tag{5}$$

The recall is defined by *RCL= ncf* / *tn*, while precision *PRC=ncf* / *nhb*, where *ncf* is the number of correctly found boundaries, *tn* is the total number of boundaries, and *nhb* is the number of hypothesized boundaries, meaning the total number of boundaries found by the segmentation module. *F* is bounded between 0 and 100, where *F*=100 is a perfect segmentation result and *F*=0 implies segmentation to be completely wrong.
Table 1 shows results for segmentation by the hybrid method.

System	RR (%)	*RCL* (%)	*PRC* (%)	*F* (%)
DSD+SLC	not applicable	66.05	40.23	50
BIC+SLC	not applicable	63.33	36.81	45.20
DSD+SLC+ HMM	78.26	75.48	57.03	64.97
BIC+SLC+ HMM	75.44	72.72	51.53	60.31
DSD+SLC+ MLC+HMM	88.53	86.36	75.41	80.51
BIC+SLC+ MLC+HMM	88.53	86.36	75.41	80.51

Table 1: Performance of the segmentation accuracies (%). SLC: segment-level clustering, MLC: model-level clustering

In our experiments, the metric-based segmentation using BIC+SLC yielded lower F-scores than DSD+SLC. The DSD method is more accurate the BIC approach in presence of short segments, while both approaches are equivalent on long segments.

The hybrid approach significantly outperforms direct metric-based segmentation. The DSD+SLC+HMM method shows better results than the BIC+SLC+HMM approach. The best segmentation results are achieved by the hybrid segmentation using model-level clustering. Both the DSD+SLC+MLC+HMM and BIC+SLC+MLC + HMM approaches provide the same segmentation results due to the model-level clustering (MLC).

4. CONCLUSION

In this paper different segmentation methods for speech stream have been compared on a panel discussion television programs. A hybrid algorithm, which combines metric-based and model-based segmentation using model-level clustering, is shown to outperform the distance metric-segmentation and a hybrid approach without model-level-clustering.

5. REFERENCES

[1] P. C. Woodland, T. Hain, S. Johnson, T. Niesler, A. Tuerk, S. Young, "Experiments in Broadcast News Transcription," *ICASSP 1998*, May 1998.

[2] M. Siegler, U. Jain, B. Raj, R. Stern, "Automatic Segmentation, classification and Clustering of Broadcast News Transcription System," *DARPA Speech Recognition workshop*, pp. 97-99, 1997.

[3] H. Gish, N. Schmidt, "Text-Independent Speaker Identification," *IEEE Signal Processing Magazine,* pp. 18-21, Oct. 1994.

[4] T. Kemp, M. Schmidt, M. Westphal, A. Waibel, "Strategies for Automatic Segmentation of Audio Data," *ICASSP 2000,* 2000.

[5] P. Yu, F. Seide, C. Ma, E. Chang, "An Improved Model-Based Speaker Segmentation System," *EUROSPEECH 2003, 2003*.

[6] T. Wu, L. Lu, K. Chen, H.-J. Zhang, "UBM-Based Real-Time speaker Segmentation for Broadcasting News," *ICME 2003,* vol.2, pp. 721-724, July 2003.

[7] S. Chen, P. Gopalakrishnan "Speaker Environment and Channel Change Detection and Clustering via the Bayesian Information Criterion," *DARPA Broadcast News Transcription and Understanding Workshop,* Feb. 1998.