FRAME PURIFICATION FOR CLUSTER COMPARISON IN SPEAKER DIARIZATION

Xavier Anguera[1,2], *Chuck Wooters*[1], *Javier Hernando*[2]

[1] International Computer Science Institute (ICSI)
1947 Center St., Suite 600, Berkeley, CA 94704, U.S.A.
[2] Technical University of Catalonia (UPC), TALP Research Group
Jordi Girona 1-3 D5, 08034 Barcelona, Spain
{xanguera,wooters}@icsi.berkeley.edu, javier@gps.tsc.upc.es

ABSTRACT

Speaker diarization is often performed as a first step to speaker or speech recognition systems, which work better when the input signal is split into its speakers. When performing speaker diarization, it is common to use an agglomerative clustering approach in which the acoustic data is first split in small pieces and then pairs are merged until reaching a stopping point. The speaker clusters often contain non-speech frames that jeopardize discrimination between speakers, creating problems when deciding which two clusters to merge and when to stop the clustering. In this paper, we present one algorithm that aims to purify the clusters, eliminating the non-discriminant frames –selected using a likelihood-based metric– when comparing two clusters.We show improvements of over 15.5% relative using three datasets from the most current Rich Transcription (RT) evaluations.

1. INTRODUCTION

The goal of speaker diarization is to segment an audio recording into speaker-homogeneous clusters [1]. This involves finding the change-points between speakers and regions of non-speech, and clustering together the regions belonging to the same speaker. Typically, this segmentation must be performed with little or no knowledge of the characteristics of the recording or of the speakers in the recording. Speaker diarization is sometimes referred to as the "Who spoke when?" task, although systems generally do not identify specific speakers by name.

The output of a speaker segmentation system can be used as a first step for multiple tasks, including speaker and speech recognition. Speech recognition is improved by adapting the models to the clustered speakers, and by performing vocal tract length normalization (VTLN) techniques on the individual speakers. In speaker recognition, a clustering into speakers is necessary to perform speaker identification or verification (e.g. for a user authentication system).

The most commonly used algorithm for speaker diarization is hierarchical agglomerative clustering [2],[3],[4],[5], including significant work applying these techniques to the meetings domain [6], [7], [8], [9]. In this approach, the signal is first divided into a number of short segments (more than the estimated number of speakers), and then the segments are iteratively merged together into speaker clusters based on their acoustic similarity. The process stops when a stopping criterion is met.

Various metrics have been proposed to measure the acoustic similarity between any two speaker clusters [10], [11]. The Bayesian Information Criterion (BIC) [12] is the metric that is most often used, which in this work is based upon the methods presented in [13]. By doing such comparison, we attempt to discriminate between clusters belonging to the same speaker and clusters belonging to different speakers. This is made more difficult by the existence of data assigned to each cluster that is common to all of them. This is the case for silence and speech frames that don't carry speaker specific information (generally referred to as non-speech frames). We call clusters containing such frames "impure".

In [14], two different algorithms are presented to purify the clusters, and therefore improve the Speaker Diarization performance. In this paper, we revisit in more detail and extend the frame-level purification, and apply it to speaker diarization using a state of the art agglomerative clustering system, where each cluster model contains a variable number of Gaussian mixtures.

By inspecting a Gaussian mixture model (GMM) with few mixtures trained on the data, we observe that the non-speech frames are normally very well modelled by a few of the Gaussian mixtures and have a high likelihood and a small variance. We show that frames with high likelihood are less likely to discriminate well between different clusters, and we study how such likelihoods are affected by the number of Gaussian mixtures assigned to each cluster model.

We run diarization experiments using the frame-purification algorithm on three datasets from the NIST's RT04s and RT05s evaluations [15], two as development sets and one as an evaluation set. We show that it improves considerably the diarization in both sets.

2. AGGLOMERATIVE SPEAKER CLUSTERING

As explained in [5] and [13], our speaker clustering system is based on agglomerative clustering. It initially splits the data into K clusters (where $K >$ number of speakers), and then iteratively merges the clusters (according to a merge metric based on ΔBIC) until a stopping criterion is met. Our clustering algorithm models the acoustic data using an ergodic hidden Markov model (HMM), where the initial number of states is equal to the initial number of clusters (K). Upon completion of the algorithm's execution, each remaining state is taken to represent a different speaker. Each state contains a set of M_D sub-states, imposing a minimum duration on the model (we use $M_D = 3$ seconds). Within the state, each of the sub-states share a probability density function (PDF) modelled via a Gaussian mixture model (GMM).

Our clustering algorithm for the meetings domain, taken as a baseline in this work, consists of the following steps:

1. Use delay-and-sum [16] to create one single "enhanced" channel from all input microphones.
2. Run speech/non-speech detection on the enhanced input data.

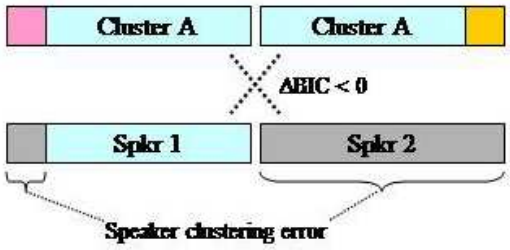

Fig. 1. *Speaker clustering errors caused by non-speech frames*

3. Extract acoustic features from the data and remove non-speech frames.
4. Determine the number of initial clusters K from the amount of data and create models for those initial clusters via linear initialization.
5. Determine the complexity (amount of Gaussian mixtures) to be assigned to each model in order to better represent the data in the cluster.
6. Perform iterative merging using the following steps:
   (a) Run a Viterbi decode to resegment the data.
   (b) Retrain the models via an Expectation-Maximization (EM) algorithm using the segmentation from step (a).
   (c) Select the cluster pair with the largest merge score (based on ΔBIC) that is > 0.0.
   (d) If no such pair of clusters is found, stop and output the current clustering.
   (e) Merge the pair of clusters found in step (c). The models for the individual clusters in the pair are replaced by a single, combined model.
   (f) Determine the model complexity for the resulting clusters.
   (g) Go to step (a).

In the meetings domain, there are several available audio channels as there are multiple microphones installed around the room. We use a variation of the delay-and-sum technique [16] to combine all data into an enhanced channel, which is then used in the speaker clustering process. This technique does not require any knowledge of the number of people or their locations, nor the locations of the microphones in the room.

In order to determine the number of initial clusters and the number of mixtures to assign to each model throughout the merging process, we define a linear relationship between the amount of data in a cluster and the number of Gaussian mixtures needed to model it. Therefore, we obtain the model complexity by

$$M_i = round(\frac{N_i}{F_{gauss}}) \quad (1)$$

The number of Gaussian mixtures to model cluster i (M_i) is determined by the number of frames belonging to that cluster (N_i) divided by a constant value (F_{gauss}) determined for all the show.

Similarly, we determine the number of initial clusters of the speaker clustering process as

$$K = \frac{N_{total}}{G_{clus}F_{gauss}} \quad (2)$$

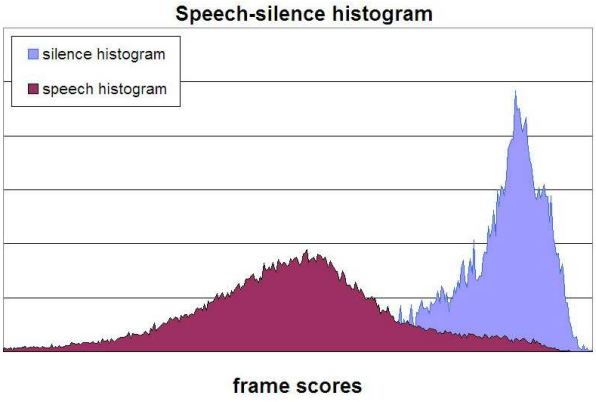

Fig. 2. *Speech-silence histogram for a full meeting*

where the number of initial clusters is a function of the amount of data we need to cluster, the number of Gaussian mixtures we want to assign per cluster (fixed as in prior work to $G_{clus} = 5$) and the constant F_{gauss}, set to be 8 seconds per Gaussian.

For the merging and clustering stopping criteria, we use a variation of the commonly used Bayesian Information Criterion (BIC) [12]. The ΔBIC compares two possible models: two clusters belonging to the same speaker or to different speakers. The variation used was introduced by Ajmera et al. [13], [17], and consists of the elimination of the tunable parameter λ by ensuring that, for any given ΔBIC comparison, the number of parameters in each model is the same.

Despite the speech/non-speech detector, some number of non-speech frames are processed by the system and assigned to a cluster, corrupting it. Furthermore, the existence of misassigned speech segments deteriorates the speaker models and increases the error rate. In the next section, we propose two algorithms to help mitigate this problem.

3. SPEAKER CLUSTER PURIFICATION AT THE FRAME LEVEL

As seen in figure 1, we can distinguish between two kinds of error caused by the existence of non-speech frames in a speaker cluster.

The first kind of errors are normally frames within a speaker's turn that do not contain any information to differentiate that speaker from others (like plosive onsets or soft fricatives) or short segments from acoustic sources other than the modelled speaker. The second kind of errors are normally segments of silence that have been missed by the speech/non-speech detector and have been labelled as part of a speaker. The algorithm presented deals with the first kind of errors (more dangerous due to the severity of the possible outcome) and is referred to as "frame level" purification.

3.1. Frame Level Purification

Due to the use of a minimum duration in the acoustic modelling, speech segments that legitimately belong to a particular cluster can be "infected" with sets of non-speech frames and frames belonging to other sources. Such sets are too short to be taken into account by the segment-based decoding or eliminated by the model-based speech/non-speech detector. However, they cause the models to diverge from their acoustic modelling targets. This is particularly important when considering whether to merge two clusters.

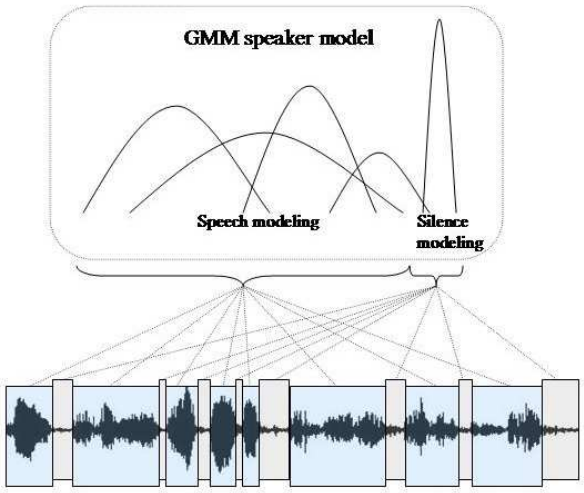

Fig. 3. *Observed assignment of frames to Gaussian mixtures*

The frame level purification presented here focuses on detecting and eliminating the non-speech frames that do not help to discriminate between speakers (e.g. short pauses, occlusive silences, low-information fricatives, etc.). Given a set of acoustic vectors X that form a speaker cluster, we can separate it into two subsets: X_1, with frames that are likely to discriminate between speakers; and X_2, non-speech frames that we wish to eliminate.

3.2. Speech and Non-Speech Modelling

Figure 2 shows the normalized histograms of the frame scores resulting from evaluating all data in a full meeting given a cluster model ($\mathcal{L}(X|\Theta_A)$) trained with speaker data. We separate the histogram in two, according to the truth file, between the speech frames and the non-speech frames. The scores of the non-speech frames are mainly located in the upper area. Some speech frames that also have a high score might be due to non-speech frames that are labelled as speech in the reference file. Even if we use a speech/non-speech detector, we have a residual error of around 5% of non-speech data that enters the clustering system. In order to purify a cluster, we need to eliminate as much of the non-speech frames as we can while maintaining the frames that discriminate between speakers.

Figure 3 illustrates a phenomenon observed when training a cluster model Θ_A, using M Gaussian mixtures, with acoustic data X. A subset (M_1) adapt their mean and variances to model the subset of speaker data (X_1), while another subset (M_2) appears to model the subset of the data (X_2) associated with non-speech information. Since the number of frames in X_1 is typically much larger than those of X_2, $|M_1| >> |M_2|$ and, at times, $|M_2|$ may be 0 if the non-speech data is minimal. Furthermore, the variance of the non-speech Gaussian mixtures in M_2 is always much smaller than M_1. Given this, any non-speech frame evaluated by the model gets a higher score than a speech frame. We take advantage of this in the frame level purification algorithm.

3.3. Frame Purification Metrics

$$\bar{\mathcal{L}}(x[i]|\Theta_A) = \frac{1}{Q}\sum_{j=-Q/2}^{Q/2-1}\sum_{m=1}^{\widetilde{M}} log\Big(W_A[m]\mathcal{N}_{A,m}(x[i+j])\Big) \quad (3)$$

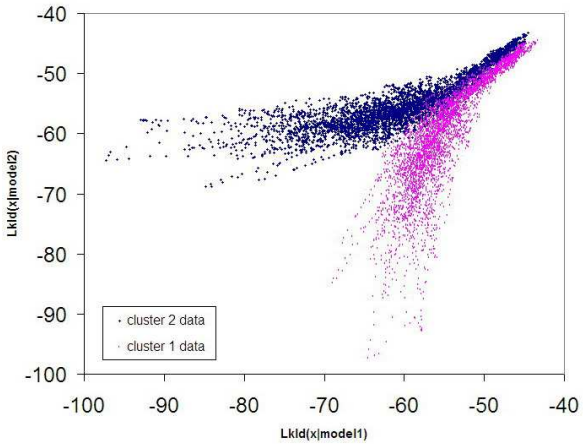

Fig. 4. *Evaluation of metric 1 on two clusters given their models*

We consider two metrics to measure this phenomenon, both using equation 3 where Q is used to average the measure around the desired value; $W_A[m]$ is the mixture weight and $\mathcal{N}_{A,m}(x[i+j])(x[\cdot])$ is the result of evaluating $x[\cdot]$ on the Gaussian mixture $\mathcal{N}_{A,m}(x[i+j])$:

Metric 1 A standard smoothed likelihood (over 100ms) of a frame, with $\widetilde{M} = M$ (all mixtures in model Θ_A) and smoothing factor $Q = 5$ (using 10ms acoustic frames).

Metric 2 The smoothed likelihood (over 100ms) on a smaller set of Gaussian mixtures that include the mixtures assigned to non-speech. We used the 50% of mixtures with smallest variance ($\widetilde{M} = M_{non-speech}$).

Figure 4 illustrates the relationship between data in two clusters according to metric 1. Both clusters are selected according to the reference file in order to contain data from only one speaker. One acoustic model is trained for each cluster's data and metric 1 is computed for each frame in each cluster according to each model. We can see that the frames with a lower value are able to distinguish very well between the two models, while the frames with higher values get all crammed together in the same area.

3.4. Implementation

When running the Speaker Diarization algorithm presented above, each cluster is modelled with a variable number of Gaussian mixtures according to the amount of data it contains. It is necessary then to take into account when we can use these metrics to purify the clusters. In figure 5, we show the histograms of speech and non-speech (according to the reference file) of metric 1 evaluated using models ranging from 1 to 8 mixtures. All model complexities have been trained with the same data and used to evaluate metric 1 on all the meeting.

As we can see, only the case of 1 Gaussian mixture shows a bigger overlap between the speech and non-speech histograms, while after 3 mixtures all plots seem identical (in fact, we ran the experiment until 20 mixtures/model with identical result). We therefore apply the frame-level purification algorithm whenever the number of gaussian mixtures is greater than 1.

The algorithm is used when gathering the data to compare two clusters using the ΔBIC metric in the following way:

1. Retrieve all frames assigned to each of two clusters and use either metric for each frame in both clusters.

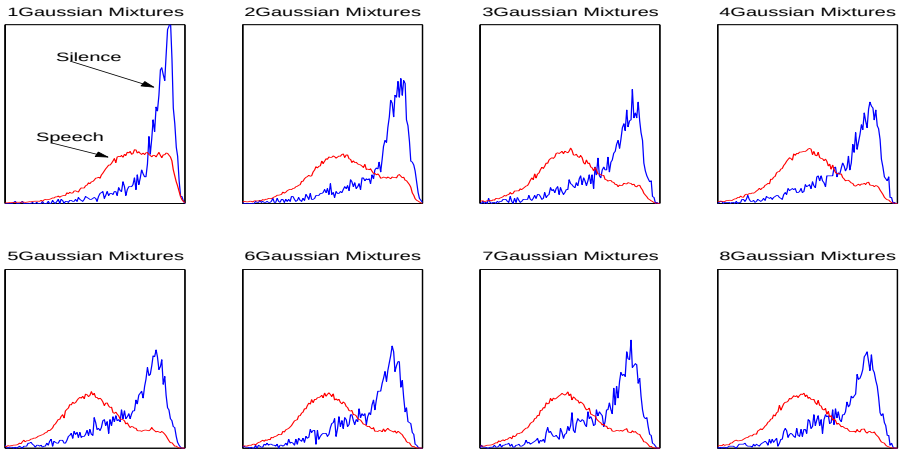

Fig. 5. *Speech/non-speech histograms for different possible model complexities*

2. If $M_i > 1$, eliminate the P% of frames in each cluster with the highest computed metric, where P is a value to be optimized according to the data.
3. Train two new models with the remaining data and use them for computing the ΔBIC metric.

4. EXPERIMENTS

We have tested the proposed algorithm using three existing datasets used in recent NIST Rich Transcription (RT) evaluations in the meetings domain [15]. The first two datasets are the RT04s evaluation and development sets, with a total of 16 meetings and an average duration of 10 minutes. We used this database as our development set. As an evaluation set, we used the RT05s set, with 10 shows and the same average characteristics as the development set. The evaluation set also has four meeting sources that did not exist in the development set. In all cases we show results on the most centrally located channel (Single Distant Microphone case, SDM) as defined by NIST.

The system performance is measured in terms of Diarization Error Rate (DER) as it is used in the NIST RT Evaluations. In computing the DER, an optimal one-to-one mapping of reference speaker identity to system output identity is performed, and the error is computed as the percentage of time that the system assigns the wrong speaker label.

For the frame purification algorithm, in figure 6, we show the DER for different possible values of the P parameter for both proposed metrics on the development set. The P indicates the percentage of frames that remain in each cluster after computing the purification metrics. This is then an indication of how strong the filtering is applied to the clusters. Both metrics converge at $P = 100\%$, in which all frames are selected for the ΔBIC comparison.

The resulting DER curves have the same shape in both cases, with a minimum for both metrics at $P = 80\%$. For all values the metric 2 outperforms the metric 1, as it is more in tune with finding the non-speech frames that need to be removed as it focuses on a subset of Gaussian mixtures smaller than the total and which include the mixtures modelling the non-speech.

Table 1 shows the DER on the evaluation and development sets. The baseline system is taken as either of both Frame-based metrics where 100% of the frames are used for the models comparison.

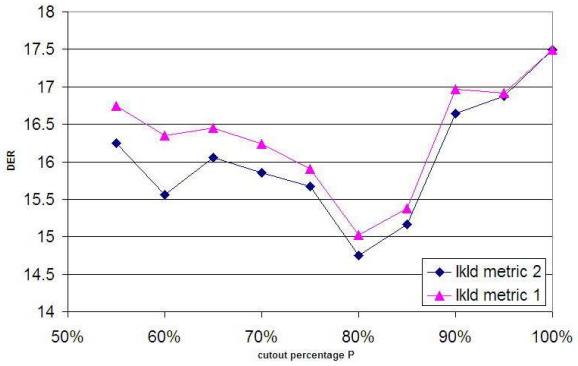

Fig. 6. *DER for different values of P for both frame-based metrics*

Purif. algorithm	Development	evaluation
baseline	17.49%	15.34%
Frame-based (metric 1)	15.02%	14.42%
Frame-based (metric 2)	**14.75%**	**14.33%**

Table 1. DER using the different metrics

From figure 6 we can see that for all values of the P% parameter we obtain an improvement over the baseline system for both metrics. This indicates that either metric successfully reflects the importance of each frame to the distinction between speakers, therefore it always improves the performance whatever the amount is that we take out.

5. CONCLUSION

In this paper, we presented and analyzed a novel technique for cluster purification in a speaker diarization system using agglomerative clustering. It detects and avoids using non-speech frames when comparing two clusters for merging or assessing the clustering stopping criterion, allowing for better discrimination of speakers. In the current work, we analyze and show results on speaker diarization, but this technique could be also used whenever we need to compare two speaker models, making it suitable for areas like speaker verification or identification. We show that this algorithm performs well

on meetings data, achieving improvements in DER of over 15.5% relative on the development set, and 6.5% relative on the evaluation set.

6. ACKNOWLEDGEMENTS

We would like to thank Adam Janin, Joe Frankel and James Fung for their help and helpful comments. This work has been done during a stay that Xavier Anguera is pursuing at ICSI, sponsored by the Spanish Ministry of Education.

7. REFERENCES

[1] D.A. Reynolds and P. Torres-Carrasquillo, "Approaches and applications of audio diarization," in *ICASSP'05*, Philadelphia, PA, March 2005, pp. 953–956.

[2] C. Barras, X. Zhu, S. Meignier, and J.-L. Gauvain, "Improving speaker diarization," in *Fall 2004 Rich Transcription Workshop (RT04)*, Palisades, NY, November 2004.

[3] D.A. Reynolds and P. Torres-Carrasquillo, "The MIT Lincoln Laboratories RT-04F diarization systems: Applications to broadcast audio and telephone conversations," in *Fall 2004 Rich Transcription Workshop (RT04)*, Palisades, NY, November 2004.

[4] R. Sinha, S. E. Tranter, J. J. F. Gales, and P. C. Woodland, "The cambridge university march 2005 speaker diarisation system," in *European Conference on Speech Communication and Technology (Interspeech)*, Lisbon, Portugal, September 2005, pp. 2437–2440.

[5] Chuck Wooters, James Fung, Barbara Peskin, and Xavier Anguera, "Towards robust speaker segmentation: The ICSI-SRI fall 2004 diarization system," in *Fall 2004 Rich Transcription Workshop (RT04)*, Palisades, NY, November 2004.

[6] Xavier Anguera, Chuck Wooters, Barbara Peskin, and Mateu Aguilo, "Robust speaker segmentation for meetings: The ICSI-SRI spring 2005 diarization system," in *Rich Transcription 2005 Spring Meeting Recognition Evaluation*, Edinburgh, Great Brittain, July 2005.

[7] Dan Istrate, Corinne Fredouille, Sylvain Meignier, Laurent Besacier, and Jean-Francois Bonastre, "NIST RT05S evaluation: Pre-processing techniques and speaker diarization on multiple microphone meetings," in *NIST 2005 Spring Rich Transcrition Evaluation Workshop*, Edinburgh, UK, July 2005.

[8] Steve Cassidy, "The macquarie speaker diarization system for RT05S," in *NIST 2005 Spring Rich Transcrition Evaluation Workshop*, Edinburgh, UK, July 2005.

[9] David van Leeuwen, "The TNO speaker diarization system system for NIST RT05s for meeting data," in *NIST 2005 Spring Rich Transcrition Evaluation Workshop*, Edinburgh, UK, July 2005.

[10] A. Solomonov, A. Mielke, M. Schmidt, and H. Gish, "Clustering speakers by their voices," in *ICASSP'98*, Seattle, USA, 1998, vol. 2, pp. 757–760.

[11] Mathew A. Siegler, Uday Jain, Bhiksha Raj, and Richard M. Stern, "Automatic segmentation, classification and clustering of broadcast news audio," in *DARPA Speech Recognition Workshop*, Chantilly, 1997, pp. 97–99.

[12] S. Shaobing Chen and P.S. Gopalakrishnan, "Speaker, environment and channel change detection and clustering via the bayesian information criterion," in *Proceedings DARPA Broadcast News Transcription and Understanding Workshop*, Virginia, USA, Feb. 1998.

[13] Jitendra Ajmera and Chuck Wooters, "A robust speaker clustering algorithm," in *ASRU'03*, US Virgin Islands, USA, Dec. 2003.

[14] Xavier Anguera, Chuck Wooters, and Javier Hernando, "Purity algorithms for speaker diarization of meetings data," in *ICASSP'06 (to appear)*, Toulouse, France, May 2006.

[15] "NIST rich transcription evaluations, website: http://www.nist.gov/speech/tests/rt," .

[16] Xavier Anguera, Chuck Wooters, and Javier Hernando, "Speaker diarization for multi-party meetings using acoustic fusion," in *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, Puerto Rico, USA, November 2005.

[17] Jitendra Ajmera, Iain McCowan, and Herve Bourlard, "Robust speaker change detection," *IEEE Signal Processing Letters*, vol. 11, no. 8, pp. 649–651, 2004.