GTTS System for the Albayzin 2010 Speaker Diarization Evaluation

Mireia Diez, Mikel Penagarikano, Amparo Varona,
Luis Javier Rodriguez-Fuentes, German Bordel

GTTS, Department of Electricity and Electronics
University of the Basque Country, Spain
mireia_diez@ehu.es

Abstract

This paper briefly describes the diarization system developed by the Software Technology Working Group (http://gtts.ehu.es) at the University of the Basque Country (EHU), for the Albayzin 2010 Speaker Diarization Evaluation. The system consists of three decoupled elements: (1) speech/non-speech segmentation; (2) acoustic change detection; and (3) clustering of speech segments. Speech/non-speech segmentation is performed by means of one of the systems presented to the Albayzin 2010 Audio Segmentation Evaluation. With the aim to detect speaker changes, speech segments are further segmented by means of a naive metric-based approach which locates the most likely spectral change points. The third element is based on a dot-scoring speaker verification system: speech segments are represented by MAP-adapted GMM zero and first order statistics, dot scoring is applied to compute a similarity measure between segments (or clusters) and finally an agglomerative clustering algorithm is applied until no pair of clusters exceeds a similarity threshold.

Index Terms: Speaker Diarization, Dot Scoring, Sufficient Statistics

1. Introduction

This paper briefly describes the dot-scoring speaker diarization system developed by the Software Technology Working Group (http://gtts.ehu.es) at the University of the Basque Country (EHU), for the Albayzin 2010 Speaker Diarization Evaluation. The system is based on three subsystems: an audio classifier developed for the Albayzin 2010 Audio Segmentation Evaluation, an acoustic change detector which was part of the system submitted to the Albayzin 2006 Speaker Tracking Evaluation [1], and a speaker verification system developed for the NIST 2010 Speaker Recognition Evaluation [2].

2. Feature Extraction

Mel-Frequency Cepstral Coefficients (MFCC) were used as acoustic features. The MFCC set, comprising 13 coefficients, including the zero (energy) coefficient, was computed in frames of 32 ms at intervals of 10 ms for the two first modules (audio segmentation and acoustic change detection). In the clustering approach, the MFCC set was computed in frames of 20 ms at intervals of 10 ms and augmented with dynamic coefficients (13 first-order and 13 second-order deltas), resulting in a 39-dimensional feature vector. Also, an energy based voice activity detector (VAD) was applied to remove those fragments (short silences) with an energy level of 30 dB (or more) under the maximum. All the speech processing computations were done by means of the Sautrela toolkit [3].

This work has been supported by the Government of the Basque Country, under program SAIOTEK (project S-PE09UN47), and the Spanish MICINN, under Plan Nacional de I+D+i (project TIN2009-07446, partially financed by FEDER funds).

3. Speech/non-speech segmentation

For this task, a simple audio segmentation system was developed, which considered five acoustic classes: (1) music, (2) clean speech, (3) speech with music in the background, (4) speech with noise in the background and (5) other (noise, long silence fragments, etc.). An ergodic Continuous Hidden Markov Model with 5 states and 512 mixtures per state was defined, using the Sautrela toolkit, under the Layered Markov Models framework [4].

The emission distributions were independently estimated for each state, applying the Baum-Welch algorithm on the corresponding sets of segments extracted from the reference segmentations of 12 development sessions. The number of mixtures per state and the transition probabilities (auto-transitions fixed to 0.999999, transitions between states and final state transitions fixed to $2 \cdot 10^{-7}$) were optimized on audio segmentation experiments over the remaining 4 development sessions. Though system performance was quite poor for the 4-class setup defined in the evaluation, when considering a 2-class speech/non-speech classification setup, the false alarm error rate was 1.16% and the miss error rate was around 1.78% for the speech class (including the three sub-classes mentioned above). Note that, since around 3% of the speech frames are mistaken, our speaker diarization error will be, at best, of that order. More details can be found in the description of the GTTS submission to the Albayzin 2010 Audio Segmentation Evaluation.

4. Acoustic change detection

Speech segments produced by the speech/non-speech detector may contain various speakers, so before clustering, a further segmentation is needed to detect speaker changes. We presented a very simple approach to detect acoustic changes (i.e. any change of speaker, background or channel conditions) in our submission to the Albayzin 2006 Speaker Tracking Evaluation (see [1] for details).

Though it was found that not only speaker changes were detected, but also many other changes, even those related to the presence of spontaneous speech events (filled pauses, coughs, etc.), the key point was that *almost all the speaker changes were detected*. Note that consecutive short segments corresponding to the same speaker can be grouped together by the clustering algorithm.

As other *metric-based* approaches (e.g. [5]), our algorithm defines and applies a metric to compare the spectral stastistics at both sides of successive points of the audio signal, and hypothesizes those boundaries whose metric values exceed a given threshold. In our approach, a kind of *normalized* crossed-BIC (XBIC) [6] is applied:

$$d(X,Y) = -\log\left(\frac{P(x|\lambda_y)P(y|\lambda_x)}{P(x|\lambda_x)P(y|\lambda_y)}\right) \quad (1)$$

The algorithm considers a sliding window W of N acoustic vectors and computes the likelihood of change at the center of that window, then moves the window n vectors ahead and repeats the process until the end of the vector sequence. To compute the likelihood of change, each window is divided in two halfs, W_l and W_r, then a Gaussian distribution (with diagonal covariance matrix) is estimated for each half and finally the cross-likelihood ratio (Eq. 1) is computed and stored as likelihood of change. This yields a sequence of cross-likelihood ratios which is post-processed to get the hypothesized segment boundaries. This involves applying a threshold τ and forcing a minimum segment size δ. In practice, a boundary t is validated when its cross-likelihood ratio exceeds τ and there is no candidate boundary with greater ratio in the interval $[t-\delta, t+\delta]$. All the parameters were heuristically optimized on the development set. The optimal values were $N = 500$, $n = 10$, $\tau = 1800$ and $\delta = 0.6$ seconds.

5. Clustering of speech segments

5.1. Gaussian Mixture Models

More than 35 hours of TV broadcast speech in Spanish, Catalan, Galician and Basque, taken from the Kalaka database [7], were used to train a gender independent GMM (Universal Background Model, UBM) consisting of 256 mixture components. Again, the Sautrela toolkit was used to estimate GMM parameters, applying binary mixture splitting, orphan mixture discarding and variance flooring.

5.2. Sufficient statistics

Zero (n) and first order (x) sufficient statistics were computed for each speech segment. The one-iteration relevance-MAP adapted and normalized mean vectors $m = \frac{\mu_{map} - \mu_{UBM}}{\sigma}$ were computed according to the following expression [8, 2]:

$$m = (\tau \mathbf{I} + diag\,(n))^{-1} \cdot x$$

5.3. Dot scoring similarity measure

Linear scoring (dot-scoring) is a simple and fast technique used in speaker verification that makes use of a linearized procedure to score test segments against target models. Given a feature stream f (the target signal) and a speaker spk, the first-order Taylor-series approximation to the GMM log-likelihood is given by:

$$\log P\,(f|spk) \approx \log P\,(f|UBM) + m_{spk}^t \cdot \nabla P(f|UBM)$$

where m_{spk} denotes the normalized mean vector of speaker spk, ∇ denotes the gradient vector with regard to the standard-deviation-normalized means of the UBM, and $\nabla P(f|UBM) = x_f$ is the first order statistics vector of the target signal f. Then, the log-likelihood ratio between the target model and the UBM, used for scoring, can be expressed as follows:

$$score\,(f, spk) = \log \frac{P\,(f|spk)}{P\,(f|UBM)} \approx m_{spk}^t \cdot x_f$$

For the diarization task, the similarity $sim(a,b)$ between two segments a and b was defined as:

$$\begin{aligned} sim\,(a,b) &= min\left\{score\,(f_a, spk_b)\,, score\,(f_b, spk_a)\right\} \\ &= min\left\{m_b^t \cdot x_a, m_a^t \cdot x_b\right\} \end{aligned}$$

5.4. Score normalization

TZ normalization was applied to dot-scores. Two development sessions were used for the estimation of T-norm and Z-norm parameters. Taking into account score normalization, the similarity measure was redefined as:

$$sim\,(a,b) = min\left\{score\,(f_a, spk_b)^{TZ}\,, score\,(f_b, spk_a)^{TZ}\right\}$$

5.5. The clustering algorithm

The similarity measure defined above was used to perform agglomerative hierarchical clustering. Given two segments (or two clusters of segments), if they are clustered together, computation of sufficient statistics for the joint cluster is straightforward:

$$\begin{aligned} x_{a+b} &= x_a + x_b \\ n_{a+b} &= n_a + n_b \end{aligned}$$

This leads to a very simple clustering algorithm:

1. **Find** $s_{max} = \underset{\forall(a,b)}{\operatorname{argmax}}\,\{sim\,(a,b)\}$
2. **If** $s_{max} < \Theta$ **then** STOP
3. **Set** $x_a = x_a + x_b$
   $n_a = n_a + n_b$
4. **Remove** cluster b
5. **Jump to** 1

Based on preliminary results on the development set, the threshold Θ was set to 3.38. Figure 1 shows system performance as a function of Θ, for the development sessions 3-16. Note that results are consistent across sessions, the optimal performance being attained for threshold values between 3 and 4.

6. Results

Table 1 shows the performance of the clustering algorithm described above on the evaluation set, using four different segmentations:

- Seg1: Reference Speaker Segmentation
- Seg2: Reference Speaker Segmentation + GTTS Acoustic Change Detection
- Seg3: Reference Acoustic Segmentation + GTTS Acoustic Change Detection
- Seg4: GTTS Acoustic Segmentation + GTTS Acoustic Change Detection

The Overall Speaker Diarization Error obtained with the Reference Speaker Segmentation (Seg1, 20.48%) would be the best performance that our clustering system could reach for the evaluation set. The difference between this result and the result obtained by the fully automated system (Seg4, 33.16%) may be explained as follows:

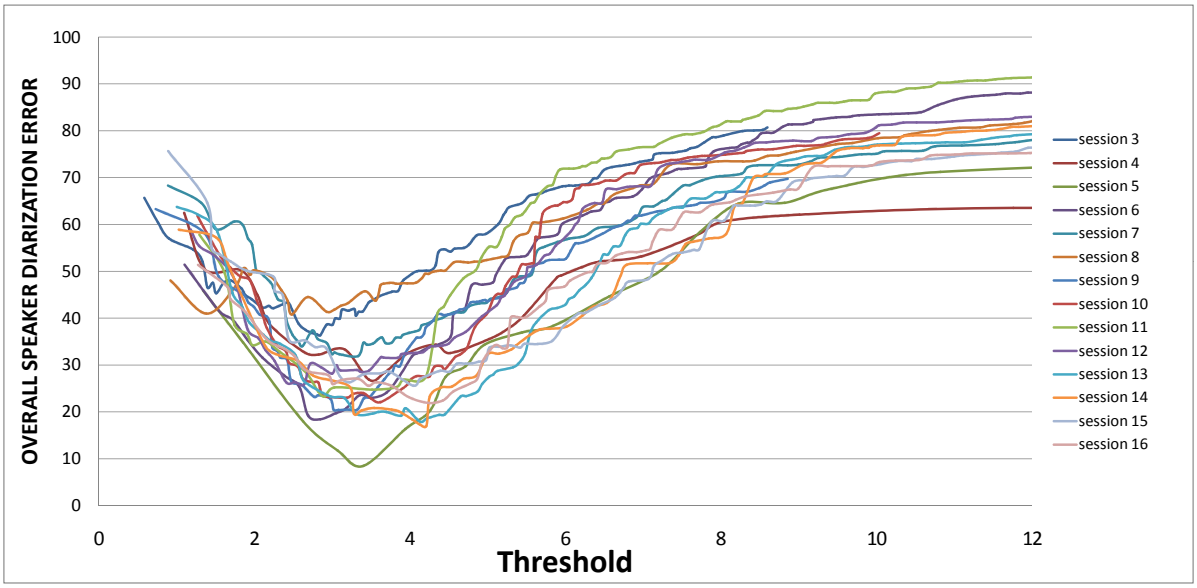

Figure 1: Overall Speaker Diarization Error as a function of the similarity threshold applied as stopping criterion in the clustering algorithm, for sessions 3-16 of the development set.

Table 1: Overall Speaker Diarization Error obtained by applying the clustering algorithm on four different segmentations of the evaluation set (see text for details).

	Seg1	**Seg2**	**Seg3**	**Seg4**
OSDErr (%)	20.48	26.14	29.61	33.16

- Difference between Seg3 and Seg4: 3.55%. Seg3 starts from a perfect audio classification, whereas Seg4 applies the GTTS audio classification system. So, the difference can be explained by the acoustic classification error.
- Difference between Seg1 and Seg2: 5.66%. Since both systems take the reference speaker segmentation as a starting point, the difference in performance can only be due to over-segmentation. Applying the acoustic change detector on the optimal speaker segmentation does not remove speaker boundaries but produces many short segments whose statistics strongly depend on local variabilities. This explains why the performance of the clustering algorithm, which is based on those statistics, degrades for short segments.
- Difference between Seg2 and Seg3: 3.47%. Seg2 includes all the speaker boundaries (plus a number of acoustic changes inside speaker turns), whereas Seg3 may be missing some of them. This explains the difference.

6.1. Processing time

Table 2 shows the CPU time (expressed as real-time factor, $\times$RT) employed in six separate operations: (1) feature extraction for segmentation; (2) audio segmentation; (3) acoustic change detection; (4) feature extraction for clustering; (5) computation of sufficient statistics; and (6) hierarchical clustering of speech segments, for both the reference speaker segmentation and the automatic segmentation. Note that the CPU time employed in clustering is almost four times higher for the automatic segmentation than for the reference segmentation, because of the different number of speech segments: 7.24 and 3.62 segments/minute, respectively. The total CPU time of the speaker diarization system is 0.2932$\times$RT.

Table 2: CPU time (real-time factor, $\times$RT) employed by the speaker diarization system modules.

	Segmentation	
	Reference	**Automatic**
Features (segmentation)	–	0.0033
Audio segmentation	–	0.0375
Acoustic change detection	–	0.1058
Features (clustering)	0.0026	
Statistics	0.0050	
Clustering	0.038	0.139

Computations were made in two servers. The first one, devoted to acoustic classification and acoustic change detection, was a Dell PowerEdge 1950, equipped with two Xeon Quad Core E5335 microprocessors at 2.0GHz (allowing 8 simultaneous threads) and 4GB of RAM. The second one, devoted to clustering, was a Dell PowerEdge R610, equipped with 2 Xeon 5550 (each featuring 4 cores) at 2.66GHz and 32GB of RAM.

7. Conclusions

This paper describes the speaker diarization system developed by the Software Technology Working Group (http://gtts.ehu.es) at the University of the Basque Country for the Albayzin 2010 Speaker Diarization Evaluation. Though quite simple in its structure, based on a chain of four uncoupled modules: audio segmentation, acoustic change detection, computation of sufficient statistics and hierarchical clustering of speech segments, the proposed system attained competitive results in the evaluation.

Experiments carried out on different segmentations showed: (1) that the lowest error rate that the clustering algorithm could attain for the evaluation set was around 20%; and (2) that over-segmentation introduced by the acoustic change detector was the main source of degradation, because the lack of robustness in the estimation of statistics for short segments. Future work may try to improve the robustness of the clustering algorithm to short segments, or alternatively, to avoid over-segmentation while keeping the detection rate of speaker boundaries.

Though not analysed in this paper, we developed an extended version of the clustering algorithm that performed speaker diarization *simultaneously* on the whole set of sessions, thus producing a single set of speaker labels. In fact, we only realized that the optimal mapping of speaker labels would be done independently for each session the day before the deadline (October 16th, 2010). The extended algorithm included a refinement stage which grouped together session clusters according to the algorithm described above, applying the same similarity threshold. We found no way of evaluating this approach because a given label corresponded to different speakers in different sessions.

8. References

[1] L. J. Rodriguez, M. Penagarikano, and G. Bordel, *A Simple but Effective Approach to Speaker Tracking in Broadcast News*, vol. LCNS 4478 of *Lecture Notes in Computer Science*, pp. 48–55. Springer Verlag, Berlin Heidelberg: Pattern Recognition and Image Analysis (IbPRIA 2007), Joan Martí, José Miguel Benedí, Ana Maria Mendonça and Joan Serrat (Eds.), 2007.

[2] M. Penagarikano, A. Varona, M. Diez., L. J. Rodriguez-Fuentes, and G. Bordel, "University of the Basque Country System for NIST 2010 Speaker Recognition Evaluation," in *Proceedings of the II Iberian SLTech Workshop*, (Vigo, Spain), November 2010.

[3] M. Penagarikano and G. Bordel, "Sautrela: A Highly Modular Open Source Speech Recognition Framework," in *Proceedings of the ASRU Workshop*, (San Juan, Puerto Rico), pp. 386–391, December 2005.

[4] M. Penagarikano and G. Bordel, "Layered Markov Models: A New Architectural Approach to Automatic Speech Recognition," in *Proceedings of the MLSP Workshop*, (São Luís,Brasil), pp. 305–314, October 2004.

[5] S. S. Chen and P. S. Gopalakrishnan, "Speaker, environment and channel change detection and clustering via the Bayesian Information Criterion," in *Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop*, (Lansdowne, Virginia, USA), February 8-11, 1998.

[6] X. Anguera, J. Hernando, and J. Anguita, "XBIC: nueva medida para segmentación de locutor hacia el indexado automático de la señal de voz," in *Actas de las Terceras Jornadas en Tecnología del Habla*, (Valencia, España), pp. 237–242, 17-19 de noviembre de 2004.

[7] L. J. Rodriguez-Fuentes, M. Penagarikano, G. Bordel, A. Varona, and M. Diez, "KALAKA: A TV Broadcast Speech Database for the Evaluation of Language Recognition Systems," in *7th International Conference on Language Resources and Evaluation*, (Valleta, Malta), 17-23 May 2010.

[8] A. Strasheim and N. Brümmer, "SUNSDV system description: NIST SRE 2008," in *NIST Speaker Recognition Evaluation Workshop Booklet*, 2008.