A simple but effective approach to speaker tracking in broadcast news

Luis Javier Rodríguez, Mikel Peñagarikano, Germán Bordel

Grupo de Trabajo en Tecnologías del Software
Departamento de Electricidad y Electrónica. Facultad de Ciencia y Tecnología.
Universidad del País Vasco. Barrio Sarriena s/n. 48940 Leioa. Spain.
e-mail: luisjavier.rodriguez@ehu.es

Abstract. The automatic transcription of broadcast news and meetings involves the segmentation, identification and tracking of speaker turns during each session, which is known as *speaker diarization*. This paper presents a simple but effective approach to a slightly different task, called *speaker tracking*, also involving audio segmentation and speaker identification, but with a subset of known speakers, which allows to estimate speaker models and to perform identification on a segment-by-segment basis. The proposed algorithm segments the audio signal in a fully unsupervised way, by locating the most likely change points from an purely acoustic point of view. Then the available speaker data are used to estimate single-Gaussian acoustic models. Finally, speaker models are used to classify the audio segments by choosing the most likely speaker or, alternatively, the *Other* category, if none of the speakers is likely enough. Despite its simplicity, the proposed approach yielded the best performance in the speaker tracking challenge organized in November 2006 by the Spanish Network on Speech Technology.

1 Introduction

The automatic transcription of broadcast news and meetings involves the segmentation, identification and tracking of speaker turns during each session, which is known as *speaker diarization* [1][2]. This task involves the segmentation of the input signal into speaker turns, advertising, music, noise and whatever other content is included in the audio file. Then, speech segments corresponding to the same speaker are clustered together and tagged with the same label. Non-speech segments are all tagged with the special label *Other*.

To measure the speaker diarization error, first the system and reference segmentations are aligned. Then, among those labels assigned by the system to any given speaker, that appearing most times is taken as the system choice and considered equivalent to the reference label. Finally, the speaker diarization error is computed as the fraction of time speakers are correctly identified. Consider the example shown in Figure 1, where not only segmentation errors but also clustering errors are illustrated. Note, for instance, that the last segment is erroneously assigned to a third speaker. After the alignment is done, the label *s01* is considered equivalent to *mm* and the label *s02* equivalent to *ft*. Finally, it is

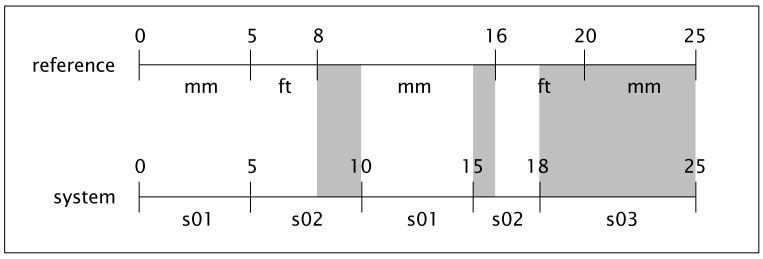

Fig. 1. An example of speaker diarization. The system provides a sequence of segments with *blind* speaker labels. After aligning the system and reference segmentations, label equivalences are set. Finally, the speaker identification error is computed as the fraction of time speakers are erroneously identified (shaded regions).

found that speakers have been erroneously identified during 10 seconds out of 25 (the shaded regions in Figure 1), which means a 40% speaker diarization error.

A slightly different task, called *speaker tracking*, is posed when speaker data are available a priori, because speaker models can be estimated and used to segment and label the audio file. Like speaker diarization, speaker tracking involves audio segmentation and speaker identification, but this latter is performed in a supervised way. In other words, the objective is to detect target speakers in a continuous audio stream. Clustering is not needed because each segment can be independently scored against speaker models and classified accordingly. Consider the example shown in Figure 2. It is close to that of Figure 1, except for the fact that the system does not provide *blind* labels, but labels of known speakers. The alignment does not determine which is the most likely mapping between reference labels and system labels. The speaker identification error is computed in a straightforward way, as the fraction of time system labels do not match reference labels. In the example of Figure 2 speakers are erroneosuly identified during 15 seconds out of 25, which means a 60% speaker identification error.

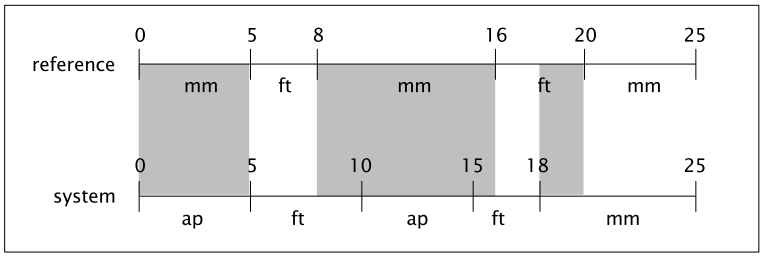

Fig. 2. An example of speaker tracking. The system provides a sequence of segments with labels of known speakers. The speaker identification error is computed as the fraction of time speakers are erroneously identified (shaded regions).

In this paper a simple approach is presented for speaker tracking in broadcast news. The segmentation step is done in a fully unsupervised way, by locating the most likely change points in the acoustic signal. Segmentation is completely

decoupled from identification and does not use speaker data. It only takes into account changes in spectral statistics. Speaker identification is done by computing the score of each segment with regard to speaker models, which are trained beforehand starting from labelled speaker data. Each segment is assigned the label of the most likely speaker or, alternatively, the label *Other*, if none of the speakers is likely enough. Note that broadcast news include music, noise, adverstising, etc. and that only a subset of the speakers is known a priori. So, under the category *Other* should fall not only non-speech segments, but also speech segments corresponding to unknown speakers.

This paper is organized as follows: in the next two sections, the audio segmentation and speaker identification algorithms are explained in detail; in section 4 the experimental setup is described, including the speech database, the audio processing and the tuning experiments; results are shown and discussed in section 5; finally, section 6 gives conclusions and tracks for future work.

2 Audio segmentation

Audio segmentation, also known as *acoustic change detection*, consists of exploring an audio file to find acoustically homogeneous segments, or, in other words, detecting any change of speaker, background or channel conditions. It is a pattern recognition problem, since it strives to find the most likely categorization of a sequence of acoustic observations, yielding the boundaries between segments as a by-product. Audio segmentation becomes useful as a preprocessing step in order to transcribe the speech content in broadcast news and meetings, because regions of different nature can be handled in a different way.

There are two basic approaches to this problem: (1) *model-based* segmentation [3], which estimates different acoustic models for a closed set of acoustic classes (e.g. noise, music, speech, etc.) and classifies the audio stream by finding the most likely sequence of models; and (2) *metric-based* segmentation [4][5][6], which defines some 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. The first approach requires the availability of enough training data to estimate the models of acoustic classes and does not generalize to unseen conditions. The second approach, also known as *blind* (unsupervised) segmentation, does not suffer from these limitations, but its performance depends highly on the metric and the threshold. Various metrics have been proposed in the literature. The most cited are the *Generalized Likelihood Ratio* (GLR) [7] and the *Bayesian Information Criterion* (BIC) [4].

Recently, the so called crossed-BIC (XBIC) [8] was introduced, improving the performance of BIC and reducing its computational cost. In this work, a kind of *normalized* XBIC is applied, a cross-likelihood metric which resembles the *Rabiner distance* [9] for the case of two multivariate Gaussians estimated from the same number of samples.

Consider two segments of speech, X and Y, of the same length, and the corresponding sequences of spectral feature vectors, $x = x_1, \ldots, x_N$ and $y = y_1, \ldots, y_N$. Assuming that the acoustic vectors are statistically independent and

that can be modelled by a multivariate Gaussian distribution, we estimate the models $\lambda_x = N(O; \mu_x, \Sigma_x)$ and $\lambda_y = N(O; \mu_y, \Sigma_y)$ and define the *dissimilarity measure* between X and Y as follows:

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

where $P(z|\lambda) = \prod_{i=1}^{N} N(z_i; \mu, \Sigma)$ is the likelihood of the acoustic sequence z given the model λ. In other words, if X and Y are acoustically close, their respective models will be quite close too, which means that $d(X,Y) \approx 0$. On the other hand, the more X and Y differ, the greater $d(X,Y)$ will become.

The audio segmentation 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 must be 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 $[t-\delta, t+\delta]$. An example of audio segmentation is shown in Figure 3.

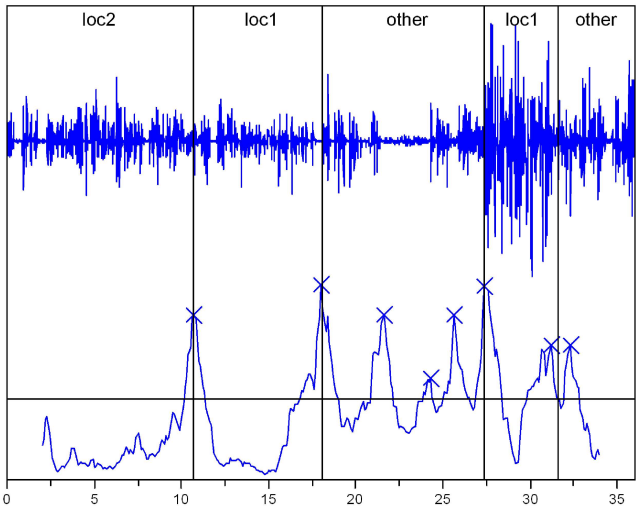

Fig. 3. An example of audio segmentation. Vertical lines represent actual boundaries, either between two speaker turns, or between a speaker turn and non-speech content. The local maxima marked with 'X' represent the boundaries hypothesized by the system.

3 Speaker identification

Once the segmentation is done, each segment must be given a speaker label or, alternatively, the special label *Other* when no speaker is likely enough. Assuming that a certain amount of training data is available for L target speakers, speaker models can be estimated beforehand. In this work, speaker models are multivariate Gaussian distributions: $\lambda_i = N(O; \mu_i, \Sigma_i)$, for $i = 1, \ldots, L$. This is just a special case of the GMM classifiers routinely used for speaker identification [10]. To classify any given segment X, firstly the *segment model* is estimated (again as a Gaussian distribution with diagonal covariance matrix) $\lambda_X = N(O; \mu_X, \sigma_X^2)$, starting from the sequence of acoustic vectors $x = x_1, \ldots, x_N$. Note that $P(x|\lambda_X) \geq P(x|\lambda_i) \quad \forall i$. The label $l(X)$ is given according to the following rule:

$$l(X) = \begin{cases} k = \arg \max\limits_{i=1,\ldots,L} P(X|\lambda_i) & \text{if} \quad \frac{1}{N} \log \left(\frac{P(x|\lambda_k)}{P(x|\lambda_X)} \right) > \epsilon \\ \\ Other & \text{otherwise} \end{cases} \tag{2}$$

where ϵ is a heuristically fixed margin which determines a threshold in the average log-likelihood ratio over which the most likely speaker k is validated as the best choice. Alternatively, if the likelihood ratio of the most likely speaker does not exceed ϵ, the label *Other* is assigned to X.

4 Experimental setup

4.1 The speech database

There was a short-term motivation for this work in the challenge for speaker tracking in broadcast news proposed in July 2006 by the Spanish Network on Speech Technologies (RTH). In fact, the experiments reported here are those carried out for that challenge, under the conditions set by the RTH [11]. The database consisted of audio tracks taken from radio broadcasts in Spanish, including many speakers, music, movie excerpts, advertising, overlaps, etc. Training data were available for 5 target speakers, consisting of 5 short utterances per speaker, 4 of them distorted with echo and reverberation. The training material for each speaker had an average length of 12.8 seconds (64 seconds all together). The test corpus consisted of 20 long tracks, with an average length of nearly 4 minutes (around 77 minutes all together). One of the training tracks, including material from only two of the target speakers, was also used for developing purposes (tuning the segmentation and identification algorithms).

4.2 Audio processing

Radio broadcasts were all sampled at 16 kHz and stored in PCM format using 16 bits per sample. The audio was analysed in frames of 25 milliseconds (400 samples) at intervals of 10 milliseconds. A Hamming window was applied and a 512-point FFT computed. The FFT amplitudes were then averaged in 24

overlapped triangular filters, with central frequencies and bandwidths defined according to the Mel scale. A Discrete Cosine Transform was finally applied to the logarithm of the filter amplitudes, obtaining 12 Mel-Frequency Cepstral Coefficients (MFCC). The choice of MFCC is based on the fact that historically there have been no features specifically designed for audio segmentation, and the MFCC are the most commonly used parameters for speaker identification.

4.3 Tuning experiments

The tuning phase consisted on running various experiments to adjust the parameters of the audio segmentation and speaker identification algorithms. As noted above, one of the audio files included in the test set, as well as the corresponding reference labels (set by human experts), were available to make the adjustments. Parameters were set to get the best match between system labels and reference labels (see Table 1). However, some considerations were taken into account beforehand, which we summarize in the following lines.

The size of the sliding window (N) should balance the performance of the segmentation algorithm for short and long segments. If N was too short, the estimation of spectral properties would focus on instantaneous events but would be less robust. If N was too long, the estimations would be robust but less sensitive to instantaneous events, and therefore very short turns would be missed. The window step (n) should be as small as possible to allow maximum resolution. However, this would increase the computational cost of the approach. The threshold for the likelihood of change (τ) should balance false alarms and missings. If τ was too low, many false boundaries would be detected; inversely, if τ was too high, some actual boundaries would be missed. However, since our objective was not an accurate segmentation but the identification of target speakers, over-segmentation did not pose a problem as long as the segments were all assigned the right speaker label. So, τ could be skewed to low values. The minimum segment size (δ) allowed to choose the most likely segment boundary in any given interval of size 2δ. If δ was too high, short segments might be missed, so it should be as small as possible, as long as it fulfils the task of avoiding *noisy boundaries* around an actual boundary. Finally, the threshold for the speaker identification likelihood (ϵ) should balance the false alarms (segments erroneously assigned to a known speaker) and missings (segments produced by known speakers and erroneously tagged as *Other*).

5 Results

To measure the performance of the proposed approach, it was used the NIST evaluation software for speaker diarization included in the Spring 2006 Rich Transcription Meeting Recognition Evaluation Plan [12]. This software takes the system labels as if they were *blind*, applying the label mapping function that minimizes the speaker diarization error, as shown in Figure 1. But what we produce are not blind but informed labels, and the speaker identification error must be measured by comparing the system and reference labels on a

Table 1. Tuned settings for the audio segmentation and speaker identification parameters: size of the sliding window (N), window step (n), threshold for the likelihood of change (τ), minimum segment size (δ) and threshold for the speaker identification likelihood (ϵ).

	Audio segmentation				Speaker identification
Parameter	N	n	τ	δ	ϵ
Tuned setting	400 (4 seconds)	10 (0.1 seconds)	1200	6 (0.6 seconds)	-1.1

frame-by-frame basis, as shown in Figure 2. To accomplish that, a little change was introduced in the NIST software, so that the score is computed as the time system labels match reference labels divided by the total audio time. Our system yields a ***17.25%*** speaker identification error, which is slightly better than that yielded by a more complex and computationally expensive system competing with ours.

Our score is comparable to other results reported in the literature [13], and is specially relevant due to the following issues:

- All the acoustic models are single Gaussians, which can hardly model the spectral variability of speakers and segments, but at the same time provide robust estimates (even when not many training data are available) and allow real-time operation of the speaker tracking system.
- Audio segmentation and speaker identification are independent modules, but further improvements might be obtained by using speaker information at the segmentation phase.
- Speaker models are estimated from a few utterances taken from radio broadcasts, many of them (80%) intentionally distorted.
- The system parameters are tuned almost blindly, using only one of the 20 audio files in the test set. More robust tuning may be accomplished if more development data were available. In particular, a ***16.26%*** speaker identification error has been obtained by tuning the parameters over the 20 audio files of the test set.

6 Conclusion

A simple approach to speaker tracking in broadcast news is presented in this paper. The audio is segmented in a fully unsupervised way, by locating the most likely change points in the acoustic signal. Speaker identification is done by computing the score of each segment with regard to speaker models, which are trained beforehand starting from labelled speaker data. All the acoustic models are single Gaussians, which provide robust estimations even when few training data are available, and allow real-time operation. The proposed system yields a ***17.25%*** speaker identification error, which is comparable to other results reported in the literature. Current work includes applying this system to a bigger database and extending its capabilities to perform speaker diarization in broadcast news and meetings.

Acknowledgments. This work has been partially funded by the Basque Government, under program SAIOTEK, projects S-PE05UN32 and S-PE05IK06. We thank to Rubén San Segundo, from the Technical University of Madrid, for his support in preparing the speech database and using the NIST evaluation software.

References

1. Tranter, S.E., Reynolds, D.A.: Speaker Diarisation for Broadcast News. Proceedings of the ISCA Speaker and Language Recognition Workshop (Odyssey 2004), pp. 337–344. Toledo, Spain. May 31 - June 3, 2004.
2. Jin, Q., Laskowsky, K., Schultz, T., Waibel, A.: Speaker Segmentation and Clustering in Meetings. Proceedings of Interspeech 2004 (International Conference on Spoken Language Processing, ICSLP), pp. 597–600. Jeju Island, South Korea. October 2004.
3. Gauvain, J.L., Lamel, L., Adda, G.: Partitioning and Transcription of Broadcast News Data. Proceedings of the International Conference on Spoken Language Processing (ICSLP'98), pp. 1335–1338. Sydney, Australia. November-December 1998.
4. Chen, S.S., Gopalakrishnan, P.S.: Speaker, Environment and Channel Change Detection and Clustering via the Bayesian Information Criterion. Proceedings of the DARPA Broadcast News Transcription and Understanding Workshop. Lansdowne, Virginia, USA. February 8-11, 1998.
5. Delacourt, P., Wellekens, C.J.: DISTBIC: A speaker-based segmentation for audio data indexing. Speech Communication 32 (2000), pp. 111–126.
6. Zhou, B., Hansen, J.H.L.: Efficient Audio Stream Segmentation via the Combined T^2 Statistic and Bayesian Information Criterion. IEEE Transactions on Speech and Audio Processing, Vol. 13, NO. 4, July 2005, pp. 467–474.
7. Gish, H., Siu, M.H., Rohlicek, R.: Segregation of Speakers for Speech Recognition and Speaker Identification. Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 1991), pp. 873–876. Toronto, Canada. May 14-17, 1991.
8. Anguera, X., Hernando, J., Anguita, J.: XBIC: nueva medida para segmentación de locutor hacia el indexado automático de la señal de voz. Actas de las Terceras Jornadas en Tecnología del Habla, pp. 237–242. Valencia, España. 17-19 de noviembre de 2004.
9. Juang, B.H., Rabiner, L.R.: A Probabilistic Distance Measure for Hidden Markov Models. AT&T Technical Journal, Vol. 64, NO. 2, pp. 391–408. February 1985.
10. Reynolds, D.A., Rose, R.C.: Robust Text-Independent Speaker Identification Using Gaussian Mixture Speaker Models. IEEE Transactions on Speech and Audio Processing, Vol. 3, NO. 1, pp. 72–83. January 1995.
11. Red Temática de Tecnologías del Habla: Propuesta de Evaluación de Sistemas ALBAYZIN-06 (Segmentación e Identificación de hablantes). IV Jornadas en Tecnología del Habla. Zaragoza, 8-10 de Noviembre de 2006. http://jth2006.unizar.es/evaluacion/albayzin06.html.
12. NIST: Spring 2006 (RT-06S) Rich Transcription Meeting Recognition Evaluation Plan. http://www.nist.gov/speech/tests/rt/rt2006/spring/.
13. Dunn, R.B., Reynolds, D.A., Quatieri, T.F.: Approaches to Speaker Detection and Tracking in Conversational Speech. Digital Signal Processing 10, pp. 93–112 (2000).