SPEAKER-BASED SEGMENTATION FOR AUDIO DATA INDEXING

Perrine Delacourt David Kryze Christian J. Wellekens

Institut EURECOM, 2229 route des Crêtes,
BP 193, 06904 Sophia Antipolis Cedex, France
delacour@eurecom.fr

ABSTRACT

In this paper, we address the problem of the speaker-based segmentation, which is the first necessary step for several indexing tasks. It consists in recognizing from their voice the sequence of people engaged in a conversation. In our context, we make no assumptions about prior knowledge of the speaker characteristics (no speaker model, no speech model, no training phase). However, we assume that people do not speak simultaneously. Our segmentation technique takes advantages of two different types of segmentation algorithms. It is organized in two passes: first, the most likely speaker changing points are detected and then, they are validated or discarded. Our algorithm is efficient to detect speaker changing points even close to one another and is thus suited for segmenting conversations containing segments of any length.

1. INTRODUCTION

With the development of telecommunications and of computer science, it is now easy to store large amounts of speech data. The problem is however how to retrieve efficiently the desired information. Therefore, automated data indexing and retrieval systems are increasingly needed.

This paper addresses the indexing via the sub-task of recognition of the sequence of speakers engaged in a conversation. In other words, the aim is to know who speaks and when. In our study, we assume that no prior information on speakers is available (no speaker or speech model, no training phase) and that people do not speak simultaneously.

This kind of indexing task could be used for example to create a database where all speeches are indexed with respect to their author or as a preliminary step in news (or movies) transcribing tasks [1], in automatic grouping speech messages [2] or in speaker tracking [3].

Our indexing task is divided in two main parts: first, the segmentation step seeks speech segments containing utterances of only one speaker. Then, the next step aims at merging speech segments related to a same speaker as described for example in [2] or [4]. In this paper, we specialize in the first step: the speaker-based segmentation.

We distinguish several types of segmentation algorithms in the literature. Segmentation algorithms based on a distance between two consecutive parts of the speech signal have been investigated in [5, 6, 7]. The problem, then, lies in the choice of a relevant threshold for distance values. A segmentation algorithm based on the Bayesian Information Criterion (BIC) is presented in [8], but proves to require long speech segments. Our segmentation technique takes advantages of these two types of segmentation techniques. First, a distance-based segmentation combined with a thresholding process as robust as possible, is operated to detect the most likely speaker changing points. Then, the Bayesian Information Criterion is used during a second pass to validate or discard the previously detected changing points.

Section 2 details two segmentation techniques: first, the algorithm using only the Bayesian Information Criterion is presented in section 2.1 and then, our segmentation technique is explained in section 2.2. Section 3 compares both techniques by evaluating their performance with criteria described in section 3.2. Results are also commented. Finally, section 4 concludes and gives possible tracks for further work.

2. SPEAKER-BASED SEGMENTATION

The aim is to segment the speech data every time a speaker change occurs. Two different types of segmentation will be reviewed: the first one relies exclusively on the BIC, referred to as the BIC procedure, and require large segments to be relevant enough. The second one we propose is based on a two step analysis: a first pass uses a distance computation to determine the changing point candidates and a second pass uses the BIC to validate or discard these candidates. Our segmentation technique shows to be less dependent of the average segment size (i.e. of the average between-speaker period).

The principle behind speaker change detection is to measure a dissimilarity value between two consecutive parts of the parameterized signal (called windows), assuming that each of these parts is related to

This work was supported by the Centre National d'Etudes des Télécommunications (CNET) under the grant n° 98 1B

one speaker only.

2.1. The Bayesian Information Criterion procedure

The first technique for dissimilarity measurement is based on the comparison of two parametric statistical models corresponding to two adjacent windows. This comparison is performed using BIC computation, proposed by Chen in [8].

The BIC is a likelihood criterion penalized by the model complexity. Given $\mathcal{X} = \{x_1, ..., x_n\}$ a sequence of $N_\mathcal{X}$ acoustic vectors, and $L(\mathcal{X}, M)$ the likelihood of $\mathcal{X}$ for the model M, the BIC value is determined by: $BIC(M) = \log L(\mathcal{X}, M) - \lambda \frac{m}{2} \log N_\mathcal{X}$, where m is the number of parameters of the model M and λ the penalty factor. We assume that $\mathcal{X}$ is generated by a multi-Gaussian process, and we consider the following hypothesis test for speaker change at time i :

- H_0: $(x_1, ..., x_{N_\mathcal{X}}) \sim \mathcal{N}(\mu_\mathcal{X}, \Sigma_\mathcal{X})$
- H_1: $(x_1, ..., x_i) \sim \mathcal{N}(\mu_{\mathcal{X}_1}, \Sigma_{\mathcal{X}_1})$ and $(x_{i+1}, ..., x_{N_\mathcal{X}}) \sim \mathcal{N}(\mu_{\mathcal{X}_2}, \Sigma_{\mathcal{X}_2})$

The maximum likelihood ratio between hypothesis H_0 (no speaker change) and H_1(speaker change at time i) is then defined by:

$$R(i) = \frac{N_\mathcal{X}}{2} \log |\Sigma_\mathcal{X}| - \frac{N_{\mathcal{X}_1}}{2} \log \left|\Sigma_{\mathcal{X}_1}\right| - \frac{N_{\mathcal{X}_2}}{2} \log \left|\Sigma_{\mathcal{X}_2}\right| \quad (1)$$

where $\Sigma_\mathcal{X}$, $\Sigma_{\mathcal{X}_1}$, and $\Sigma_{\mathcal{X}_2}$ are respectively the covariance matrices of the complete sequence, of the subset $\{x_1, ..., x_i\}$, and of the subset $\{x_{i+1}, ..., x_{N_\mathcal{X}}\}$, and $N_\mathcal{X}$, $N_{\mathcal{X}_1}$, and $N_{\mathcal{X}_2}$, are respectively the number of acoustic vectors in the complete sequence, in the subset $\{x_1, ..., x_i\}$, and in the subset $\{x_{i+1}, ..., x_{N_\mathcal{X}}\}$. The variations of the BIC value between the two models (one Gaussian versus two different Gaussians) is then given by:

$$\Delta BIC(i) = -R(i) + \lambda P \quad (2)$$

where the penalty is given by $P = \frac{1}{2}(p + \frac{1}{2}p(p + 1)) \log N_\mathcal{X}$, p being the dimension of the acoustic space, and λ is the penalty factor. A negative value of Δ-BIC(i) indicates that the two multi-Gaussian models best fit the data $\mathcal{X}$, which means that a change of speaker occurred at time i.

BIC values computation is costly, and therefore the algorithm implementation has to be done in three steps to avoid computation overload, as described in [9]:

1. A first pass is performed to determine the approximate location of the changing points. The Δ-BIC value is computed between two adjacent windows $[a, b]$ and $[b, c]$, where the boundaries a and c are fixed, and where b takes its values in $[a, c]$ and is increased at each iteration by a certain resolution step. The distance $d(a, c)$ is increased when no negative value is found for Δ-BIC. When a negative value is found, the changing point becomes the new value for a.

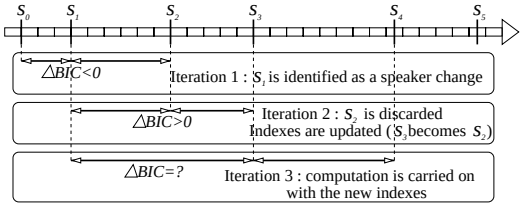

Figure 1: Principle of BIC final pass

2. The second pass uses the same method for refining the results of the first pass: the exploration intervals $[a, c]$ are chosen much smaller, and centered around the points previously selected as candidates.

3. The third pass validates the results of the second pass. If $\{s_1, ..., s_N\}$ is the set of speaker change candidates found in step 2, a Δ-BIC value is computed for each pair of windows $[s_{i-1}, s_i]$ $[s_i, s_{i+1}]$. If the value is negative, a speaker change is identified at time i. If not, the point s_i is discarded from the candidate set, so that the Δ-BIC value is now computed for the new pair of windows $[s_{i-1}, s_{i+1}]$ $[s_{i+1}, s_{i+2}$ (with the old indexes), as shown in figure 1.

This method, which consists in merging segments as long as positive values for BIC are found, is necessary for a correct estimation of the Gaussian parameters, since the model accuracy depends highly on the amount of available information. Thus, the reliability of the results is a function of the length of the sequence of acoustic vectors used for computation.

A direct consequence is that the use of the BIC algorithm alone for the speaker segmentation is not adapted for small sized segments. Indeed, the algorithm can not detect two speaker changes closer to one another than the second pass window size, which is of about 2 s. Even with larger segments, if the frequency of speaker change is too high and does not allow good parametric estimations, the algorithm yields bad results.

Another problem comes from the tuning of the penalty factor λ, which showed to be quite dependent on the type of analyzed data. In Chen's work, λ is set to 1 but in our experiments the empirical factor λ took its values between 1.2 and 1.8.

We will therefore use a more robust technique based on distance computation for the first pass, and keep the BIC algorithm for refinement in a second pass.

2.2. Our segmentation technique

The first pass of our segmentation technique relies on a distance-based segmentation. The applied measure function has to reflect how similar two adjacent segments are. A high value should indicate a change of

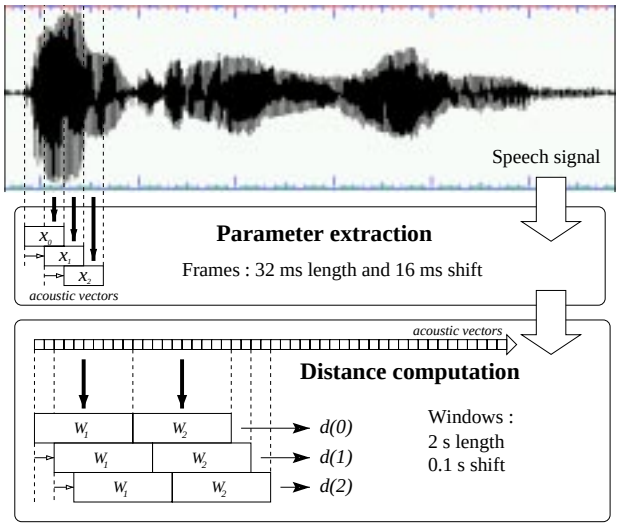

Figure 2: Distance computation process

speaker, whereas low values should signify that the two portions of signal correspond to the same speaker. Among the many possible different measures commonly used (see [5, 6, 7, 10]), the measure derived from the Generalized Likelihood Ratio (GLR) presented in [11, 12] proved to be the most efficient one, showing high and narrow peaks at speaker change, and low variation amplitude within single speaker segments (see also [13] for a study of the different measures mentioned above).

Using the same notations as in section 2.1, the likelihood ratio between the hypothesis H_0 and H_1 is defined by:

$$\lambda = \frac{L(\mathcal{X}, \mathcal{N}(\mu_{\mathcal{X}}, \Sigma_{\mathcal{X}})}{L(\mathcal{X}_1, \mathcal{N}(\mu_{\mathcal{X}_1}, \Sigma_{\mathcal{X}_1}).L(\mathcal{X}_2, N(\mu_{\mathcal{X}_2}, \Sigma_{\mathcal{X}_2})} \quad (3)$$

The GLR distance is computed by taking the logarithm of the previous expression:

$$d_{\mathrm{GLR}} = -\log \lambda$$

The GLR distance is computed for a pair of adjacent windows of the same size (about 2s), and the windows are then shifted by a fixed step (about 0.1s) along the whole parameterized speech signal. This process (see figure 2) gives as output the graph of distance with respect to time.

The GLR computation relies on a similar approach as the one used in BIC computation, but there is no penalty against the complexity of the models. Thus, a threshold δ will have to be introduced to decide whether or not the considered distance value reflects a speaker changing point or not: a measure value greater than δ will indicate a speaker change.

However, this threshold is strongly dependent on the type of analyzed data and has shown high variability with respect to recording conditions. In fact, to each test database was related a different threshold. In order to design a more robust system irrespective of the type of data, an adaptive threshold technique has to be used.

Once all distance values have been computed, a low-pass filter is applied to remove all high frequency perturbations on the graph. Then, all the "significant" local maxima are searched. A local maximum is regarded as "significant" when the differences between its value and those of the minima surrounding it are above a certain threshold (calculated as a fraction of the graph variance), and when there is no greater local maximum in its vicinity. Thus, the selection of the local maxima is not done considering the absolute value of the peaks, but rather by considering the "form factor" of the peaks. This type of detection meets the following requirements:

- It does not depend on the type of speech data (TV news, phone conversations, studio)
- The emphasize is placed on minimizing the *deletion errors* (not detecting any changing point where there is one) rather than the *insertion errors* (detecting speaker changes where they do not exist), as the created sub-segments will be likely to be merged during the second pass.

Even with a fine tuning of the detection parameters, the number of insertion errors remains high after the distance-based segmentation. A second pass using the BIC is required to merge the segments corresponding to the same speaker, and thereby to decrease the number of insertion errors.

The second pass is the exact copy of the third pass of the BIC analysis presented in section 2.1. A Δ-BIC value is computed for each changing point candidate to validate the result of the first pass. The value of the empirical factor λ has to be tuned in order to minimize the number of insertion errors without adding new deletion errors. The use of the BIC is now much more appropriate as the length of the considered segments is large enough for a good parameter estimation.

3. EXPERIMENTATIONS

3.1. Data

To test our approach, we use different types of speech data :

- 2 conversations which are artificially created by concatenating sentences of 2 s on average from the TIMIT database (clean speech, short segments), referred to as *timit2* and *timit3*.
- 2 conversations created by concatenating sentences of 1 to 3 s from a French language database (clean speech, short segments), referred to as *file1* and *file2*.

- 3 TV news broadcasts extracted from the database of the "Institut National de l'Audiovisuel" (INA) in French language (segments of any length), referred to as *extrait4*, *extrait8* and *extrait10*.
- 3 phone conversations extracted from the SWITCHBOARD ([14]) database (segments of any length, spontaneous speech), referred to as *sw2005*, *sw2007* and *sw2008*.

The speech signal is parameterized with 12 mel-cepstral coefficients. The addition of the Δ-coefficients (first derivatives) does not improve the results and increases the time of computation. For this reason, the Δ-coefficients are not used (see [13]).

3.2. Evaluation methods

A good segmentation should provide the correct speaker changes and therefore segments containing one speaker only. We distinguish two types of errors related to speaker change detection. An *insertion error* occurs when a speaker change is detected although it does not exist. A *deletion error* occurs when the process does not detect an existing speaker change.

Depending on the stage following the segmentation, these two types of errors do not have the same importance. Our segmentation technique has been designed to be embedded in a complete indexing process. In this case, insertion errors (resulting in an over-segmentation) are less critical than deletion errors. The segmentation stage will be followed by a clustering stage to group segments related to a same speaker. Thus, the insertion errors will be corrected during this next stage.

A reference segmentation is required for using this kind of error definitions. However, its accuracy, when the reference segmentation exists, could be very low since the perception of speaker changes is very subjective. One can circumvent this difficulty by defining accuracy windows around reference and detected changing points. A detected changing point is an insertion error if no reference changing point is found in the surrounding window. At the opposite, the absence of a changing point candidate in a window around a reference changing point corresponds to a deletion. The use of such accuracy windows does not lead to realistic error evaluation: indeed, the width of the window should depend on the speaking rate, but also on the semantic context of the conversation. The ultimate test is performed by listening the segments in isolation and deciding upon the quality of their ending point detection.

3.3. Results

In order to evaluate our segmentation technique, we compare it with the BIC procedure, described section 2.1. For both techniques, we count the number of insertion errors and the number of deletion errors.

	BIC		1^{rst} pass		2^{nd} pass	
	I	D	I	D	I	D
extrait4 (23 pts)	8	2	24	2	9	2
extrait8 (22 pts)	2	0	9	0	3	0
extrait10 (38 pts)	10	8	18	7	7	7

Table 1: French TV news: insertion (I) and deletion (D) errors respectively with the BIC procedure, the first pass and the second pass of our algorithm

In all the tables of results, we mention for each speech file the total number of speaker changing points in brackets. The second and third columns indicate respectively the number of insertion errors (I) and the number of deletion errors (D) for the BIC procedure. The next two columns concern the first pass and the last two columns the second pass of our segmentation technique. For each pass, the first column represents the number of insertion errors (I) and the second one the number of deletion errors (D), as for the BIC procedure.

For both segmentation techniques, the parameters they involved are set up for each database. The longer the speaker segments are, the higher parameter λ (involved in the BIC) should be. Likewise, the longer the segments are, the larger the windows to compute the distance or the BIC should be. One can also notice that parameters are not influenced by the language: parameters of both segmentation techniques used with American and French synthetic conversations (both built with shorts sentences) are quite the same. The small differences are probably due to the recording conditions.

By examining table 1, we can see that the number of insertion and deletion errors respectively for the BIC procedure and for the second pass applied to French TV news are comparable. Performances of both techniques are equivalent. We can also notice that the number of insertions errors is significantly reduced between the first and the second pass. In fact, most of the insertion errors, occurring during the first pass, are due to speaker intonation changes related to the semantic content of the sentence or environment changes.

Concerning the conversations built with the TIMIT sentences (see table 2), a significant reduction of the number of deletion errors is observed between our segmentation technique and the BIC procedure, as expected. Thus, our segmentation technique proves to be more efficient than the BIC procedure when applied to conversations where speaker changing points are close to one another. However, one can notice that if the second pass improves the number of insertions errors, at the same time, the number of deletion errors becomes worse.

The same remarks can be made concerning the con-

	BIC		1^{rst} pass		2^{nd} pass	
	I	D	I	D	I	D
timit2 (29 pts)	15	7	14	4	11	6
timit3 (27 pts)	11	10	25	4	11	5

Table 2: TIMIT conversations: insertion (I) and deletion (D) errors respectively with the BIC procedure, the first pass and the second pass of our algorithm

	BIC		1^{rst} pass		2^{nd} pass	
	I	D	I	D	I	D
file1 (21 pts)	4	11	8	6	7	7
file2 (21 pts)	3	10	2	1	2	2

Table 3: French conversations : insertion (I) and deletion (D) errors respectively with the BIC procedure, the first pass and the second pass of our algorithm

versations built with the French short sentences (see table 3). In both cases, and more generally with conversations containing only small segments, our segmentation technique could be reduced to the first pass (i.e. to the distance-based segmentation) to be more efficient according to the number of deletion errors.

The phone conversations (see table 4) present some particularities compared to other data, which may impair the segmentation and the evaluation processes. Indeed, while one person is speaking, the other person may interrupt or speak at the same time to pronounce small words or interjections like "Yeah" or "Hum-hum" to agree on what is said. Spontaneous speech presents similar characteristics. When these small words are uttered while the other person is speaking, our hypothesis is not respected. These words also degrade the segmentation process since they are too small to be correctly detected. Also depending on the context of the segmentation, they may be not relevant. For instance, to know that speaker X has intervened for 0.3 seconds to say "Yeah" does not have any significance for our indexing task. At the opposite, if the accuracy level required for a transcription task is very high, then it becomes necessary to correctly detect these small words. In our context, we decide not to take them into account. That explains the high numbers of insertion errors for the phone conversations (table 4), particularly for the first pass of our segmentation technique. Indeed, the distance-based segmentation is more sensitive to every change (intonation or environment) than the BIC-based segmentation. However, the distance-based segmentation only detects the beginning or the end of these small words, but not both boundaries. Depending on phone conversations, results are sometimes better with the BIC-procedure, sometimes better with our segmentation technique.

	BIC		1^{rst} pass		2^{nd} pass	
	I	D	I	D	I	D
sw2005 (19 pts)	10	6	41	6	17	7
sw2007 (66 pts)	20	13	31	17	18	17
sw2008 (30 pts)	1	13	6	9	3	7

Table 4: SWITCHBOARD phone conversations: insertion (I) and deletion (D) errors respectively with the BIC procedure, the first pass and the second pass of our algorithm

4. CONCLUSION AND FURTHER WORK

In this paper, we proposed a speaker segmentation technique, composed of a distance-based algorithm followed by a BIC-based algorithm. This segmentation technique proves to be as efficient as the BIC procedure in the case of conversations containing long segments and to give better results than the BIC procedure when applied to conversations containing short segments. Our efforts will now concentrate on combining this segmentation stage with the merging stage to form the complete indexing process (i.e. the recognition of the sequence of speakers engaged in a conversation). At this point, recognition results obtained at the end of the complete indexing process will allow to fully validate our segmentation technique.

5. REFERENCES

[1] P. Woodland and al., "The development of the 1996 HTK broadcast news transcription system," in *DARPA speech recognition workshop*, 1997.

[2] D. Reynolds and al., "Blind clustering of speech utterances based on speaker and language characteristics," in *ICSLP98*, 1998.

[3] A. E. Rosenberg and al., "Speaker detection in broadcast speech databases," in *ICSLP98*, 1998.

[4] J. Johnson and P. Woodland, "Speaker clustering using direct maximisation of the MLLR-adapted likelihood," in *ICSLP98*, 1998.

[5] M. A. Siegler and al., "Automatic segmentation, classification, and clustering of broadcast news audio," in *DARPA speech recognition workshop*, 1997.

[6] C. Montacié and M.-J. Caraty, "Sound channel video indexing," in *Eurospeech*, pp. 2359–2362, 1997.

[7] H. Beigi and S. Maes, "Speaker, channel and environment change detection," in *World congress of automation*, 1998.

[8] S. Chen and P. Gopalakrishnan, “Speaker, environment and channel change detection and clustering via the Bayesian information criterion,” in *DARPA speech recognition workshop*, 1998.

[9] A. Tritschler, “A segmentation-enabled speech recognition application using the BIC criterion,” Master’s thesis, Institut EURECOM, France, 1998.

[10] F. Bimbot and al., “Second order statistical measures for text-independant speaker identification,” *Speech communication*, vol. 17, Aug 1995.

[11] H. Gish, M.-H. Siu, and R. Rohlicek, “Segregation of speakers for speech recognition and speaker identification,” in *ICASSP*, pp. 873–876, 1991.

[12] H. Gish and N. Schmidt, “Text-independent speaker identification,” *IEEE signal processing magazine*, oct. 1994.

[13] P. Delacourt and C. J. Wellekens, “Audio data indexing: use of second-order statistics for speaker-based segmentation,” in *ICMCS*, 1999. accepted for publication in ICMCS99.

[14] J. Godfrey and al., “SWITCHBOARD: telephone speech corpus for research and development,” in *ICASSP*, 1992.