A metric-based approach to speaker change detection

Petra Zochová, Vlasta Radová

University of West Bohemia in Pilsen, Department of Cybernetics
pzochova@kky.zcu.cz, radova@kky.zcu.cz

Abstract: The paper deals with the problem of automatic speaker change detection. A new metric-based algorithm, called AlgBICMap algorithm, is proposed in this paper. The AlgBICMap algorithm allows to create a map of BIC (Bayessian Information Criterion), which enables us to detect efficiently fields of speech of individual speakers. In comparison with a typical metric-based approach, the advantage of the proposed algorithm is its robustness because it uses more information, not only information provided by adjacent windows. In addition to that, the AlgBICMap algorithm can be used also for speaker tracking tasks.

1 Introduction

The aim of the automatic speaker change detection is to extract homogeneous segments containing the longest possible utterances produced by a single speaker. Many efforts have been devoted to this problem in the last years, mainly due to the large number of possible applications, e.g. in speaker recognition systems, in automatic transcribing tasks, or as an improvement of speech recognition systems.

There are three main speaker change detection approaches [1]. In the *metric-based approach* the speaker changes are determined as the moments in which a distance measure computed between two adjacent windows shifted along the whole speech signal reaches a local maximum. In the *model-based approach* it is assumed that a model of each speaker, the voice of which is contained in the utterance, has been trained before the speaker change detection algorithm starts. The speaker changes are then detected as the instants when it is necessary to change the speaker model in order it matches the speech signal. The last approach is the *decoder-guided approach*. Here, the speaker changes are determined according to information provided by a speech recognition system which decodes the spoken audio stream at first (e.g. possible speaker changes are at every silence location).

In this paper, we are interested in the metric-based approach, because it does not require any other information or things except the speech signal itself (i.e. neither speaker model, nor speech recognizer). We have improved the metric-based approach in order to get more information about the changes. When a distance is computed only between two adjacent windows, a lot of pieces of information will be lost. It is as if we wear blinkers and our eyes can look straight ahead only; then we have no information from the left side or the right side. For that reason we have decided to compute not only a *graph* of distances (which is created by the distances computed between two adjacent windows shifted along the whole speech signal), but the whole *map* of distances. In such a case, the distance is computed between any two windows in the speech signal and the distances are depicted. We are interested in ”reading from the map” in order to obtain as much information as possible.

The organization of the paper is as follows: First, in Section 2, the AlgBICMap algorithm is explained. Then in Section 3 a possible use of the proposed algorithm in a speaker tracking task is briefly introduced. In Section 4 experiments are described and achieved results are presented. Finally, in Section 5, some conclusions are given.

2 AlgBICMap algorithm

The AlgBICMap algorithm detects speaker changes in a speech record. It uses the Bayessian Information Criterion (BIC) in order to form a BIC-map.

2.1 Preprocessing

The first phase of our algorithm is preprocessing when silent parts are removed from the record using the algorithm described in [2]. Then vectors of 12 mel-frequency cepstral coefficients (MFCCs) are computed from the speech stream not containing silence. The vectors of MFCCs are then used as feature vectors.

2.2 BIC computation

In speaker change detection algorithms the ΔBIC value (1) is commonly computed for two adjacent windows which are shifted along the whole speech signal. A speaker change is detected if the ΔBIC value for the actual adjacent windows is negative [1]. The ΔBIC value is given by

$$\Delta\text{BIC} = -R + \lambda P, \tag{1}$$

where

$$R = \frac{N}{2}\log|\Sigma| - \frac{N_1}{2}\log|\Sigma_1| - \frac{N_2}{2}\log|\Sigma_2|\,, \tag{2}$$

λ is a penalty factor, and

$$P = \frac{1}{2}\left(d + \frac{1}{2}d\,(d+1)\right)\log N. \tag{3}$$

N_1 and Σ_1 are respectively the number and the covariance matrix of the feature vectors in the window W_1. Similarly, N_2 and Σ_2 are the number and the covariance matrix of the feature vectors, respectively, in the window W_2. Further, $N = N_1 + N_2$, Σ is the covariance matrix of the feature vectors of both windows, and d is the dimension of the feature vectors. The value of the empirical factor λ has to be tuned in order to reduce the number of false alarms without increasing the number of missed detections.

2.3 BIC-map computation

The ΔBICvalue (1) is computed between the actual window (starting from the beginning of the speech stream) and all the following windows (the map is symmetric, so we do not need to compute the distance between the actual window and the previous windows). Then we move on to the next window and the same process is repeated until the end of the speech stream is reached. The distances are stored in a matrix called the BIC-map.

2.4 Binary BIC-map

When we use a zero threshold, we can transform the BIC-map naturally into a binary BIC-map with the values 0 and 1. If a value of the distance in the BIC-map is positive, the corresponding value in the binary BIC-map is set to 1, otherwise it is set to 0. An example of the binary BIC-map is presented in Fig. 1. The black color represents the value 1 and the white color represents the value 0.

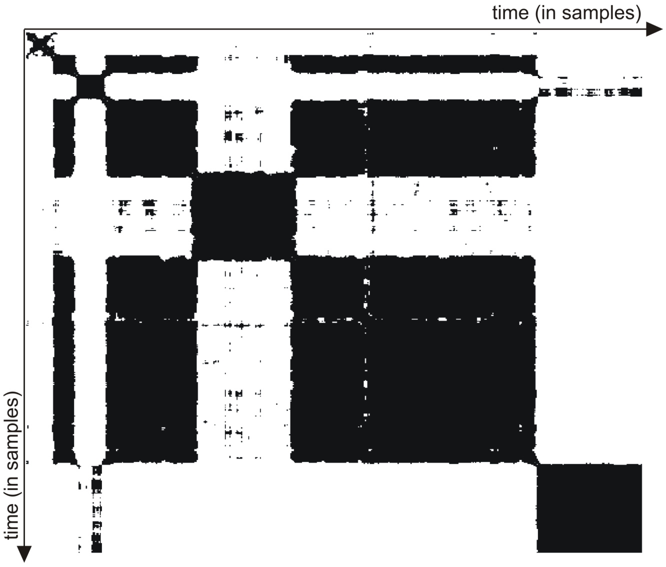

Figure 1: An example of the binary BIC-map.

A lot of interesting things can be seen from Fig. 1. We can see for example, that the main diagonal of the binary BIC-map is scattered with square-shape elements. We call them *diagonal squares*. One diagonal square represents speech of one speaker. The pass between two squares represents a speaker change.

From the BIC-map we can also trace all the segments of the speech stream where a particular speaker speaks. When we chose a diagonal square in the binary BIC-map and find all black rectangle-shape elements placed above and below that square in the vertical direction, the segments on the vertical axis corresponding to the rectangle-shape elements represent the segments where the same speaker speaks.

2.5 Detection of the diagonal squares

The algorithm consists of three phases. The aim of these phases is to detect the bottom-right and upper-left corners of the diagonal squares, because the corners represent potential speaker changes.

From now on it will be supposed that the BIC-map can be expressed as a matrix

$$M(i,j),\ i = 1 \ldots m,\ j = 1 \ldots m, \tag{4}$$

where m represents the length of the speech stream, i denotes a row of the BIC-map, j is a column of the BIC-map, $M(i,j) = 1$ for the black points of the BIC-map, and $M(i,j) = 0$ for the white points.

First phase: VERTICAL EXAMINATION

The aim of the first phase is to find the bottom-right corners of the diagonal squares. We have used the following steps for this purpose:

Step 1: Initialization – start at the upper-left corner of the BIC-map (i.e. set $i = 1$ and $j = 1$) and set the counter of the potential speaker changes to 1 (i.e. $c = 1$).

Step 2: Compute the ratio

$$ratio_v = \frac{\sum_{k=i}^{j} M(k, j)}{j - i + 1} 100\% \,, \tag{5}$$

which is the percentage of the black points (i.e. the points with the value 1) in the column j between the rows i and j.

Step 3: If $ratio_v$ is lower than a percentage threshold $t_{vertical}$, there is a potential speaker change in time $j - 1$ (in samples). Then set $T_v(c) = j - 1$, $i = j$, and $c = c + 1$.

Step 4: Set $j = j + 1$.

Step 5: Repeat Steps 2, 3, and 4 until the end of the BIC-map is reached (i.e. until $i = m$ and $j = m$).

Second phase: HORIZONTAL VALIDATION

The aim of the second phase is to find the upper-left corners of the diagonal squares. In the ideal case the upper-left corner of a diagonal square is equal to the bottom-right corner of the previous diagonal square. The first upper-left corner is supposed to be at the beginning (upper-left corner) of the BIC-map. In this phase only several rows around the potential speaker changes found in the first phase are examined.

Step 1: Initialization – start with the first potential speaker change found in the first phase (i.e. set $c = 1$) and set $i = T_v(c) - n$, where n is an experimentally determined constant.

Step 2: Compute the ratio

$$ratio_h = \frac{\sum_{k=i}^{T_v(c)} M(i, k)}{T_v(c) - i + 1} \tag{6}$$

which is the percentage of the black points (i.e. the points with the value 1) in the row i between the columns i and $T_v(c)$.

Step 3: If $ratio_h$ is higher than a percentage threshold $t_{horizontal}$, there is a potential speaker change in time i (in samples). Then set $T_h(c) = i$, $c = c + 1$, and $i = T_v(c) - n$. Go to Step 5.

Step 4: Set $i = i + 1$.

Step 5: Repeat Steps 2, 3, and 4 until the end of the BIC-map is reached (i.e. until $i = m$ and $j = m$).

Third phase: ACCURATE POSITION OF THE SPEAKER CHANGE

We have 2 values for each potential speaker change: one from the first phase, and the other from the second phase (see Table 1 for illustration). The aim of the third phase is to find the accurate positions of the speaker changes.

potential speaker change order	1	2	3	...
potential speaker change found in the 1st phase	$T_v(1)$	$T_v(2)$	$T_v(3)$	...
potential speaker change found in the 2nd phase	$T_h(1)$	$T_h(2)$	$T_h(3)$	...

Table 1: *Table of potential changes.*

Suppose that the accurate position of the c-th speaker change is investigated and denote the position $T(c)$.

Step 1: Form 2 adjacent windows: The first window starts at the time $T(c-1)$,

$$T(c-1) = \max\{T_v(c-1),\, T_h(c-1)\}, \tag{7}$$

and ends at the time T; the second window start at the time T and ends at the time $T(c+1)$,

$$T(c+1) = \min\{T_v(c+1),\, T_h(c+1)\}. \tag{8}$$

Step 2: Let $T_{\text{min}}(c) = \min\{T_v(c),\, T_h(c)\}$ and $T_{\text{max}}(c) = \max\{T_v(c),\, T_h(c)\}$. For T from T_{min} to T_{max} compute the ΔBIC value (1) between the windows formed in Step 1. The T for which the ΔBIC value is the lowest is the sought position $T(c)$.

2.6 Diagonal belt

Computation of the whole BIC-map is very time demanding. For a speaker change detection task, we do not need to compute the whole map. It is sufficient to compute only a belt around the diagonal of the map (see Fig. 2). However, the algorithm of the diagonal squares detection has to be adapted accordingly.

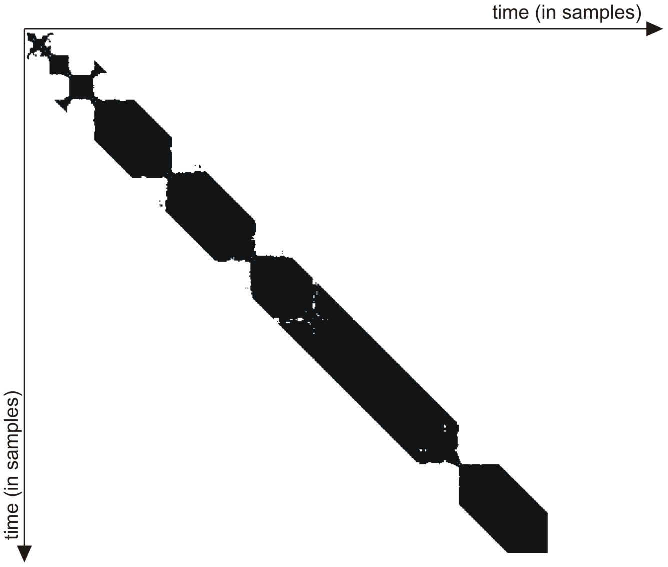

Figure 2: Diagonal belt for the BIC-map in Fig. 1.

3 Speaker tracking

As it was already mentioned in Section 2.4, speech segments of the same speaker can be easily detected in the BIC-map.

Suppose we have a speech stream and using the AlgBICMap algorithm described above we have found that a speaker started to speak at the time s_1 and finished his speech at the time instant e_1. Another segment containing speech of only one speaker starts at s_2 and ends at e_2. The task is to determine, whether in these two segments the same speaker speaks.

In the BIC-map we locate the rectangle with the corners $[s_2, s_1], [s_2, e_1], [e_2, s_1], [e_2, e_1]$. The relative number of the black points (the points with the value 1) in the rectangle is counted. A high value of the relative number means that the same speaker speaks in both segments.

4 Experiments and results

In our experiments, the length of the windows, for which the ΔBIC value (1) is computed, was 2 s, and the window shift was 0.25 s. The penalty factor λ was set to 1.8. The threshold $t_{vertical}$ was set to 80 %, and the threshold $t_{horizontal}$ was set to 70 %. The constant n in the second phase of the algorithm for the detection of the diagonal squares was equal to 7. Width of the diagonal belt was 10 seconds.

The AlgBICMap algorithm was tested with two different types of records: TV news and radio news.

- The radio news test set consisted of 8 records containing news broadcasted by the Czech radio station Český rozhlas 2 – Praha. The length of each record was about 10 minutes, each record contained about 23 speaker changes on average. Speakers in the records did not speak simultaneously and the interval between two consecutive speaker changes was quite long.
- The TV news test set contained 3 records of newscasts of different Czech TV channels. The length of the records ranged from 11 to 20 minutes, each record contained about 94 speaker changes on average. Similarly as in the radio news, the speakers did not speak simultaneously. However, unlike the radio news, about 9 % of speaker changes were quite close (less than 2 s).

Two types of errors can happen during the speaker change detection. A false alarm (FA) occurs when a speaker change is detected, although it does not exist. On the contrary, a missed detection (MD) occurs when the algorithm does not detect an existing speaker change. If we know the number of FA and MD for a record, we can determine the accuracy and the false alarm rate (FAR) that were achieved for the record using a speaker change detection algorithm. The accuracy is defined as

$$\text{Accuracy} = \frac{\text{number of true speaker changes} - \text{number of MD}}{\text{number of true speaker changes}} \, 100\% \,, \tag{9}$$

and the FAR is determined according to the formula

$$\text{FAR} = \frac{\text{number of FA}}{\text{number of true speaker changes} + \text{number of FA}} \, 100\% \,. \tag{10}$$

In Tables 2 and 3 results of the AlgBICMap algorithm introduced in this paper are given and compared with the MDistBIC algorithm described in [3]. The results demonstrate an increase of accuracy and a significant reduction of FAR when the AlgBICMap algorithm was used.

	AlgBICMap algorithm		MDISTBIC algorithm	
record	accuracy	FAR	accuracy	FAR
022119MN	90.74%	12.90%	85.46%	36.05%
022219MN	90.32%	11.43%	84.13%	19.23%
022819CN	89.09%	14.06%	87.93%	32.56%
mean	90.05%	12.80%	85.84%	29.28%

Table 2: *Speaker change detection results achieved for records of TV news.*

	AlgBICMap algorithm		MDISTBIC algorithm	
record	accuracy	FAR	accuracy	FAR
020323	100.00%	25.00%	100.00%	50.00%
021523	95.65%	17.86%	86.96%	30.30%
021623	92.59%	30.77%	100.00%	46.00%
021723	100.00%	21.62%	100.00%	40.82%
022223	95.00%	16.67%	100.00%	40.63%
022323	100.00%	14.71%	100.00%	42.00%
022423	100.00%	41.94%	100.00%	53.85%
022923	96.77%	24.71%	93.33%	28.57%
mean	97.50%	24.16%	97.54%	41.52%

Table 3: *Speaker change detection results achieved for records of radio news.*

5 Conclusion

The algorithm proposed in this paper represents a new way of speaker change detection. As the experimental results show, the use of the BIC-map allows to improve the performance of the speaker change detection. The algorithm is more robust and reduces significantly the number of false alarms. In addition, when a speaker speaks more than once in a recording, we are able to recognize it from the map and detect all the segments where the speaker speaks. The BIC-map can be therefore used also in speaker tracking tasks.

6 Acknowledgements

The work was supported by the Grant Agency of the Czech Republic, project no. 102/05/0278.

References

[1] DELACOURT P., WELLEKENS C.J.: ”DistBIC: A speaker-based segmentation for audio data indexing.”, *Speech Communication, vol. 32, pp. 111–136, 2000*

[2] ZOCHOVÁ P., RADOVÁ V.: "A Method for Speaker-Based Segmentation of Audio Signals.", *Speech Processing, 13th Czech-German Workshop, Prague, Czech Republic, pp. 109–112, 2004*

[3] ZOCHOVÁ P., RADOVÁ V.: "Modified DISTBIC algorithm for speaker change detection.", *Interspeech 2005, Lisbon, Portugal*