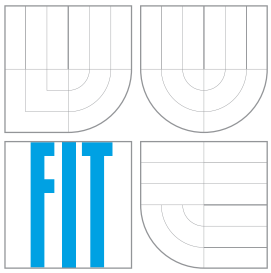
VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÉ GRAFIKY A MULTIMÉDIÍ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER GRAPHICS AND MULTIMEDIA

DIARIZACE MEETINGOVÉ ŘEČI - KDO MLUVÍ KDY?

MEETING SPEECH DIARIZATION - WHO SPEAKS WHEN?

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE — RADOVAN TŮMA
AUTHOR

VEDOUCÍ PRÁCE — Ing. MARTIN KARAFIÁT
SUPERVISOR

BRNO 2007

Abstract

This work is trying to propose Diarization System based on Bayesian Information Criterion (BIC). In this paper is possible to find description of background theory and short description of previously used systems. Idea of this work is to try to use methods proposed earlier in a faster and more reliable way. Proposed system was tested on some records to prove its error rate. Results of tests are not very good but some possible improvements are proposed.

Keywords

segmentation clustering BIC Diarization

Abstrakt

Tato prace obsahuje návrh diarizačního systému. Systém je postaven na bázi BIC (Bayesian Information Criterion). Ve zprávě naleznete stručný popis dříve vyvíjených systémů a stručnou teorii popisující, jak by diarizační systém měl pracovat. Práce se zohlednit dřívejích prací jiných autorů a výsledkem by měl být systém s některými vylepeními. Vzlepení se zaměřují zejména na rychlejí segmentaci s minimální ztrátou přesnosti a co nepřsnější clustering

Klíčová slova

segmentace clustering BIC Diarizace

Citace

Radovan Tůma: Meeting Speech Diarization - Who speaks when?, diplomová práce, Brno, FIT VUT v Brně, 2007

Meeting Speech Diarization - Who speaks when?

Prohlášení

Prohlauji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Martina Karafiáta

.....
Radovan Tůma
22nd May 2007

Contents

1 Introduction **2**
1.1 Motivation . . . 2

2 Background theory **3**
2.1 Parametrization . . . 3
2.2 BIC . . . 4
2.3 Robust Diarization System Model . . . 5
2.3.1 Speech Activity Detection . . . 5
2.3.2 Segmentation . . . 5
2.3.3 Clustering . . . 5
2.3.4 Iterative Resegmentation . . . 7

3 Previous works **8**

4 Segmentation **9**
4.1 Segmentation via BIC . . . 9
4.2 Segmentation via Combination of T^2 and BIC . . . 9
4.3 Some experiments with BIC based methods . . . 10
4.4 Faster BIC Segmentation . . . 13
4.4.1 Updating Covariance Matrices Faster . . . 13
4.4.2 Algorithm Description . . . 15
4.4.3 Variants of Optimized Algorithm . . . 15
4.5 Tests on Synthetic Data . . . 17

5 Clustering **18**

6 System Description **20**
6.1 Speech Activity Detection . . . 20
6.2 Parametrization . . . 21
6.3 Segmentation . . . 21
6.4 Clustering . . . 21

7 Test **23**
7.1 Test description . . . 23
7.2 Test results . . . 23

8 Conclusion **24**

Chapter 1

Introduction

In many systems like a system for automatic text transcription or speaker recognition it is important to find speech segments with independent speakers. It is called speaker diarization. One very good method was proposed by Chen and Gopalakrishnan, this method was based on Bayesian Information Criterion(BIC). In this work, I am trying to improve the speed of this method for segmentation of record and to add clustering part which will be better than BIC based clustering methods proposed earlier. In this paper diarization system will be proposed. Paper is divided into several chapter describing background theory about BIC, segmentation algorithm and clustering. In last chapter is description of tests and their results.

1.1 Motivation

The main motivation to create a new diarization to be used on Faculty of Information Technology at Brno University of Technology. Diarization systems are very useful for many applications. Their main purpose in research area is to help in automatized gathering data for training and testing other speech processing systems. In the real word it can be used to achieve better results in automatic speech transcription which can be used for searching keywords in speech. Transcription systems are also very helpful for deaf people.

Chapter 2

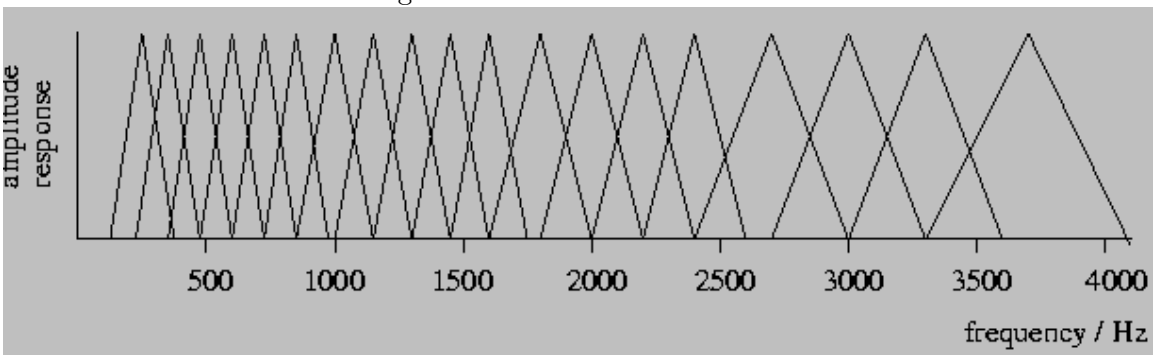
Background theory

This chapter is introduction into the theory of speech parametrization and to the Bayesian Information Criterion.

2.1 Parametrization

Typically we can't see much useful information directly from waveform so we need some method to describe an audio signal in better way. It is most common to use information about frequencies during the short period of time. We use so called feature vectors. As in many other works, I have decided to use Mel Frequency Cepstral Coefficients(MFCC). MFCC is based on Mel-Cepstrum introduced by Davies and Mermelstein [2]. Main advantage of this feature set is that mel cepstrum reflects different importance of different frequencies. MFCCs are computed using short analysis window where discrete Short Time Fourier Transform is computed and mel-scale filter bank is applied on magnitude spectra. Filters follow the mel-scale where center frequencies and band edges are linear for low frequency (<1 kHz) and then logarithmically increasing as is shown on the picture 2.1. The feature sets based on mel cepstrum differs in count of filters in filter bank, in the way in which filter ban is constructed and in count of mel coefficients. The delta and delta delta (acceleration) of MFCC, energy or logarithm of energy are often added to the vector. There exists many studies about feature vector selection but their results are often very different so it is very hard to say which combination of features is the best.

Figure 2.1: mel-scale filter bank.



mel scale filter bank used by Davies and Mermelstein [2]

2.2 BIC

Informations in this section are taken mainly from [1] and [10]. Bayesian Information Criterion(BIC) is a statistical criterion for model selection based on likehood of models and penalty for the complexity of model. I don't want to go into details about this criterion because it can be easily found in many other papers or in statistical literature. BIC criterion is used to choose the best of competiting models. In speech segmentation we are working with two models. First model expects that whole segment of length N belongs to one speaker. Second models the same data as being from 2 defferent speekears and expect changing point b. Second model (consisting of 2 parts) is usualy describing segment better and it is why the BIC is using penalization. Having 2 competiting models we can compute delta BIC value for this models and see which is better. It is explained in detail in equations 2.3, 2.4 nad 2.5. Usually there is used Gaussian process as a model, so we are computing likelihood from covariance matrices. Gaussian process is described by mean values(μ) and covariance matrix(Σ).

$$\mu = \frac{1}{N}\sum_{i=1}^{N} x_i \tag{2.1}$$

$$\Sigma = \frac{1}{N}\sum_{i=1}^{N}(x_i - \mu)(x_i - \mu)' \tag{2.2}$$

and formal model definition looks like:

$$M_1 : X = x_1, x_2, ..., x_N \sim N(\mu, \Sigma)$$

$$M_2 : X = x_1, x_2, ..., x_b \sim N(\mu_1, \Sigma_1); x_{b+1}, x_{b+2}, ..., x_N \sim N(\mu_2, \Sigma_2)$$

where $x_i \in X, i \in 1..N$ is a feature vector.
It can be seen that we have two concurrent models first expects that segment is homogeneous and the second expects changing point on position b. Now we compute BIC values:

$$BIC(M_1) = -\frac{d}{2}Nlog2\pi - \frac{N}{2}log|\Sigma| - \frac{N}{2} - \frac{1}{2}\lambda(d + \frac{1}{2})d(d+1))logN \tag{2.3}$$

$$BIC(M_2) = -\frac{d}{2}Nlog2\pi - \frac{b}{2}log|\Sigma_1| - \frac{N-b}{2}log|\Sigma_2| - \frac{1}{2}\lambda(d + \frac{1}{2})d(d+1))logN \tag{2.4}$$

From above equations 2.3 and 2.4 we can compute difference between the competing models which tell us if there is a changing point around point b.

$$\Delta BIC(b) = \frac{1}{2}(Nlog|\Sigma| - blog|\Sigma_1| - (N-b)log|\Sigma_2|) - \frac{1}{2}\lambda(d + \frac{1}{2})d(d+1))logN \tag{2.5}$$

where part $-\frac{1}{2}\lambda(d + \frac{1}{2})d(d+1))logN$ could be seen as a threshold. λ is a parametr which influences the penalty for model complexity. The proper choice of λ value will lead to the correct segmentation. Too large value will cause missing of changepoints and too small will lead to find changepoints where they should't be. $\Delta BIC(b) > 0$ means that there is a changing point around b and this changing point should be in position b where $\Delta BIC(b)$ is maximal.
Problem of this method is to find correct value of λ and to find a proper length of segment

for computation. Because estimation of covariance matrices could be very noisy if we don't have at least $2d$ of feature vectors. If too many parameters are used we could have more than 2 speakers in models and this could lead to improper segmentation decision.
Because we need a relatively large amount of data for good matrix estimation this method is not very reliable for short segments of speech. Some research was done about how noisy the estimation is for given dimension of feature vector using BIC [3]. It was shown that BIC prefers segments of similar length and that the minimal length for proper decision is quite large. It's best to have at least 100 vectors if we use feature vectors with about 30 coefficients to make a good decision. BIC is going to have very low error rate for about 200 vectors and more in each model.

2.3 Robust Diarization System Model

Now I would like to introduce how should robust diarization system looks. It can be best seen from the picture 2.3. Diarization system consists from several parts. Speech detection is used to detect a speech in a record. It is especially important if record was made under noisy conditions. BIC change detection finds points where speakers alternates, it's based on Bayesian Information Criterion as described in the previous chapter. Full covariance agglomerative clustering merges the leafs of tree representing clusters until the stopping criterion is met. Iterative re-segmentation comes after clustering phase. It means that models of speakers are made according to the computed clusters and small segments from original record are compared with this models and added to the correspondent clusters. This step is repeated up to 3 times. This work is focused mainly on the parts dealing with segmentation and clustering.

2.3.1 Speech Activity Detection

It is very important to have good Speech activity detection (sometimes called Voice Activity Detection - VAD) because outputs from BIC segmentation and clustering are bad if they are influenced significantly by noise. My first idea was to make detection using simple algorithm based on zero crossing rate and energy in a short window (about 10ms). This idea was bad because testing data are noisy very often. So it is better to use more sophisticated method. It proved useful to involve speech/non-speech discrimination using Hidden Markov Models to distinguish between voice and other classes like music, silence or noise.

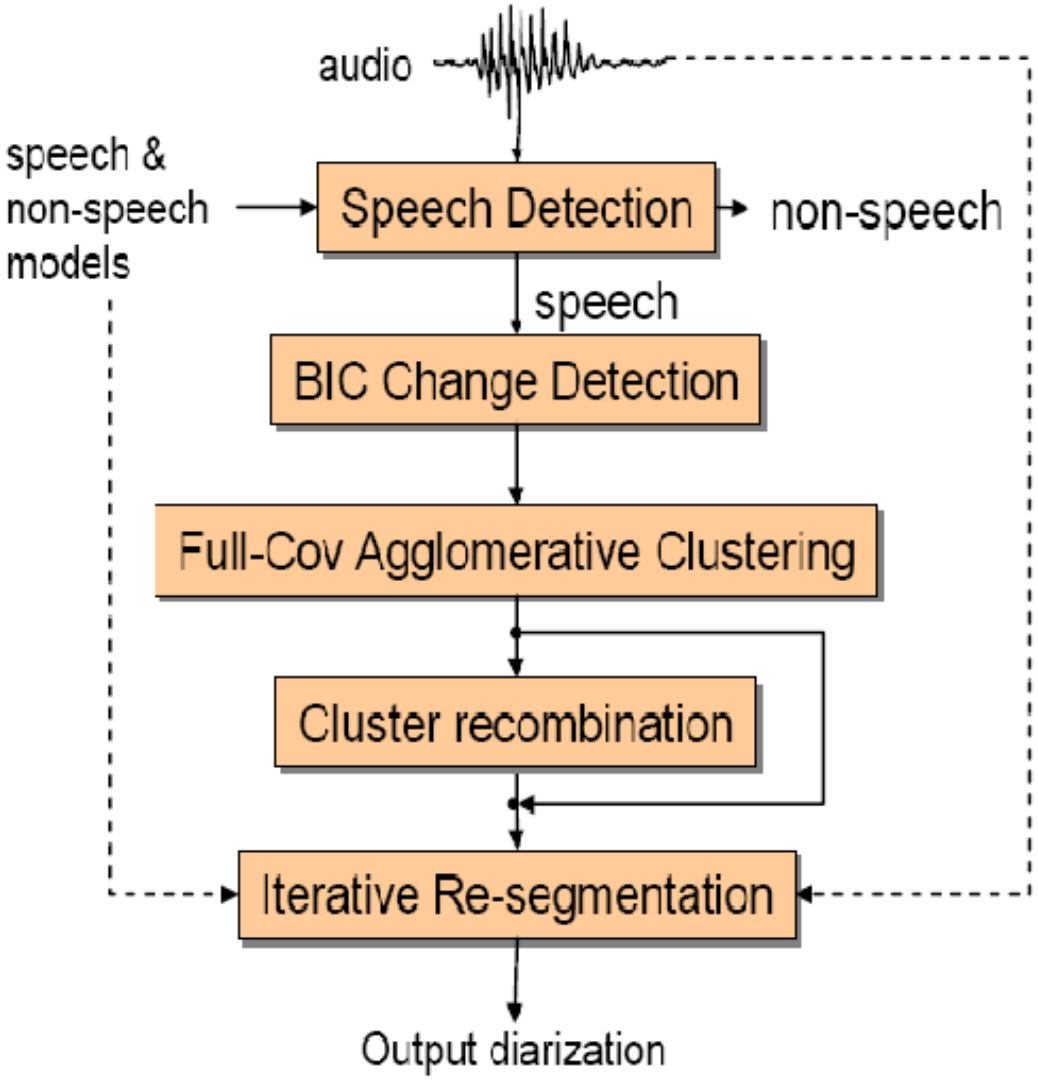
2.3.2 Segmentation

In this thesis speech segmentation refers to the problem of finding points where speakers are changing. In previous works many different approaches were tested and BIC segmentation as was described in the work of Chen and Gopalakrishnan [1] proved itself to be a good solution. It is possible to use BIC based method in several different ways. Some implementations use it directly to find changing points [1], others use it in a combination of some candidate change points preselection where BIC is used only to confirm the decision [10].

2.3.3 Clustering

The choice and results of different clustering methods depends a lot on error rates of segmentation and speech detection. Another important factor in the choice of clustering algorithm

Figure 2.2: Robust diarization system graph



picture from [6]

is the knowledge about the quantity of speech of each single speaker and the length of segments. In the general purpose diarization system we can't expect the long segments and we can't expect large quantity of data for individual speakers. This reasons are why we need to choose a simple model. As it was shown in some papers, the full covariance matrix models with BIC can be used for clustering [5]. Many works also uses gaussian mixture models(GMM) [8] but it is not very good idea because GMM are very poorly estimated for short segments.

2.3.4 Iterative Resegmentation

The idea of this step is to train models corresponding to the cluster obtained by previous steps and to segment record according to this models. The main prerequisite for this step is that good cluster purity was reached in previous steps. Resegmentation should correct bad clustering decisions comming from short segments earlier. This step is repeated usually up to 3 times.

Chapter 3

Previous works

It is important as a first step in new system development to have some knowledge about "state of art" to know what the others are using and what methods have good results. My work have been inspired especially by work of Chen and Gopalakrishnan [1]. They are using BIC criterion for speech segmentation with good results but the algorithm is very slow and computational expensive. Some improvement was proposed by Zhou and Hansen [10]. Zhou and Hansen are using T^2 as a first criterion because it needs only means of feature vector and it leads to the less computation. They involve BIC only for decision if changing point was chosen correctly by T^2. This two works were most important for my design of segmentation algorithm.
I have also studied some works dealing with clustering problem like the diarization system proposed by Leeuw [5] who is using BIC segmentation similar to [1] and clustering based also on BIC. He is using agglomerative clustering where he is merging 2 clusters with best value of ΔBIC. In this work is used similar method but with some improvements to make it more reliable.

Chapter 4

Segmentation

This chapter will describe details of some BIC segmentation algorithms and algorithm chosen for my work will be explained.

4.1 Segmentation via BIC

Now I would like to give a short description of original method based on BIC as proposed by Chen and Gopalakrishnan in their paper "Speaker, environment and channel change detection and clustering via the Bayesian information criterion" [1]. In this work they are trying to find changing points in audio channel using equation 2.5. Algorithm starts in taking 1 second window at the beginning of an audio stream, then they try to compute $\Delta BIC(b)$ for each b inside this window. If they find values bigger than zero they label the point b as the changing point and continues with a new window beginning in b+1. If they don't find any b for which $\Delta BIC(b) > 0$ they extends window by 1 s and starts the computation of $\Delta BIC(b)$ again.
This method is very slow because it needs to compute covariance matrices for each point but results of this method are very good if you have enough data for good estimation of matrices. It means that initial 1 s window could be used for 12-dimensional feature vector(like classic Mel Frequency Cepstral Coefficients). In this case the method should work well for $b = S + 24..E - 24$ where S is the start point of window and E is the end point of window. If you want to use feature vector with more dimension is good to make initial window longer.
One of the weakness of this method is also in large difference of size of the left part and right part of the model 2, some papers are discussing this problem and its look like that it could influence results a lot especially for short windows [3].

4.2 Segmentation via Combination of T^2 and BIC

This method is very similar to the previous one but authors used T^2 for finding candidate changing points and then they use ΔBIC to make segmentation decision.
The algorithm is based on T^2 statistic value in point b given by equation:

$$\Delta T^2(b) = \frac{b(N-b)}{N}(\mu_1 - \mu_2)'\Sigma^{-1}(\mu_1 - \mu_2) \tag{4.1}$$

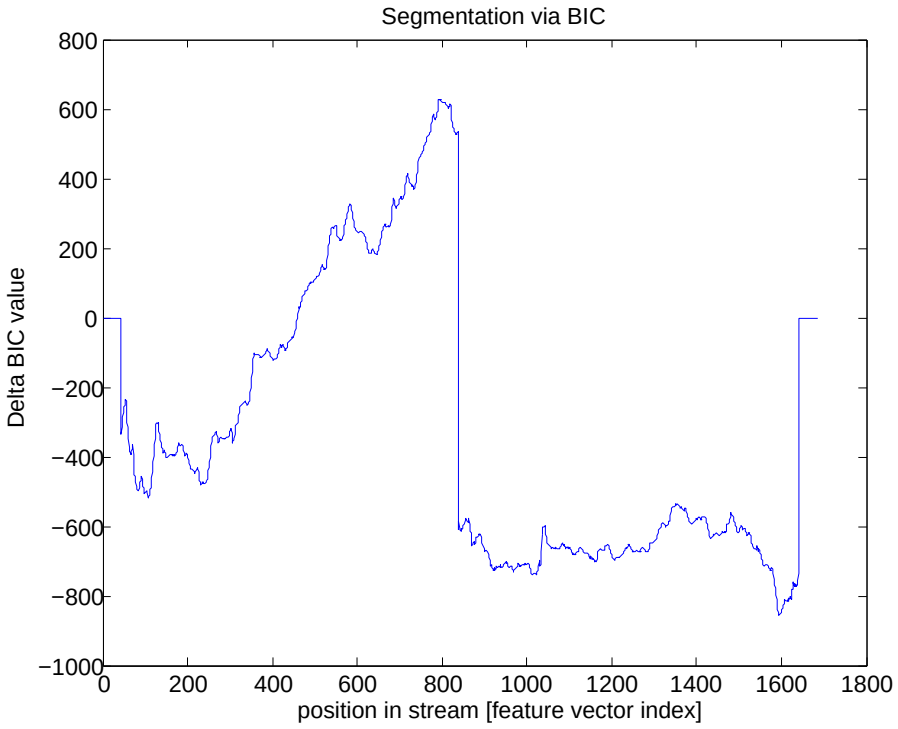
where N is the count of features vectors, b is candidate changing point and μ_1, μ_2 and Σ are means and covariance matrices of models as described on page 4 This method is much

faster than method described in previous section because you need to compute covariance matrices for model 2 only for candidate changing point b preselected by maximal value of equation 4.1.

4.3 Some experiments with BIC based methods

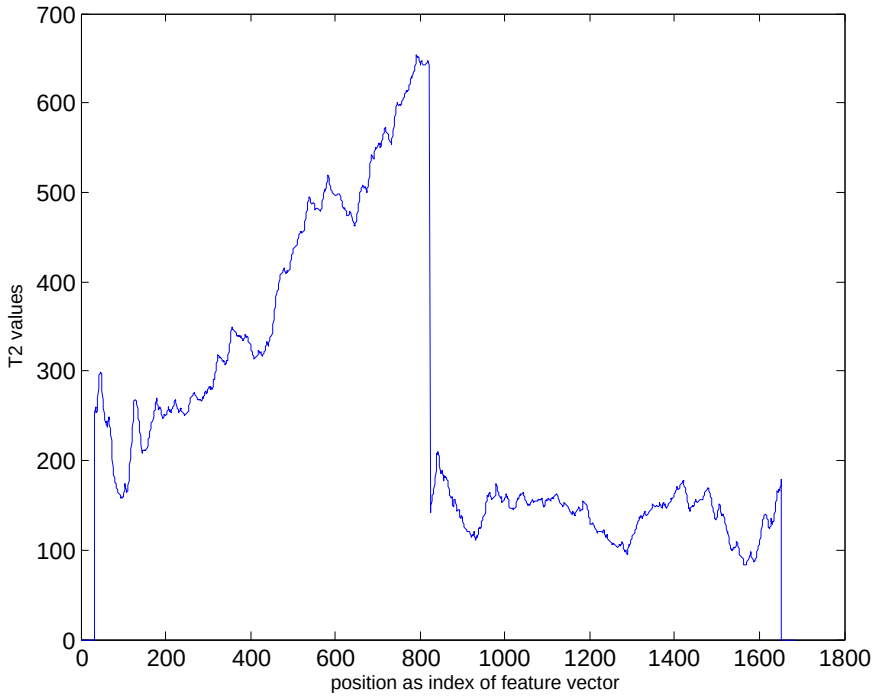
Figures with curves showing the distance function in the given part of record around speaker change will be shown.

Figure 4.1: Chen and Gopalakrishnan: Segmentation via BIC.



On this picture are ΔBIC values for audio stream with speaker change at point 819. Algorithm has found changing point 795 which is close to the right one.

Figure 4.2: Hansen and Zhou: segmentation via T^2 and BIC.



On this picture are T^2 values for audio stream with speaker change at point 819. Changing point was found on position 792.

On both figures above we could see that algorithms are working with offset (zero values in leftmost and rightmost regions) and that they are starting to compute a new window after finding changing point. It can be seen as the vertical line right to the maximal value. This vertical line is to the right of max value because of there is offset like in the beginning and end of the stream. This offset is used to prevent noisy estimation of covariance matrices because of insufficient data.
It is also very interesting to see the ΔBIC values inside of the file computed in the fixed size sliding window. Which can be described as:

$$\Delta BIC(b) = \frac{1}{2}(Nlog|\Sigma| - \frac{N}{2}log|\Sigma_1| - \frac{N}{2}log|\Sigma_2|) - \frac{1}{2}\lambda(d + \frac{1}{2})d(d+1))logN \tag{4.2}$$

where the Σ is for model 1 as described on page 4 and Σ_1 and Σ_2 are computed for model 2 such as the point b is in the middle of the window.

4.4 Faster BIC Segmentation

Later in my work I have tried to make faster implementation of the original BIC segmentation method. For efficiency improvement it is essential to avoid computing of two full covariance matrices for each point inside window. This can be done by several means like using T^2 described above.

4.4.1 Updating Covariance Matrices Faster

I have decided to try another approach. It is based on the first algorithm and the improvement is based on faster estimation of covariance matrices. The idea is that the means for computation of covariance matrix are changing slowly and so we can add a new feature vector to the covariance matrix or remove it with no need to recomputing means. So the first covariance matrices and means for window will be computed traditionally as in equation 2.1 and 2.2. So the first μ_1, Σ_1 and μ_2, Σ_2 will be:

$$\mu_1 = \frac{1}{b}\sum_{i=1}^{b} x_i \tag{4.3}$$

$$\Sigma_1 = \frac{1}{b}\sum_{i=1}^{b}(x_i - \mu_1)(x_i - \mu_1)' \tag{4.4}$$

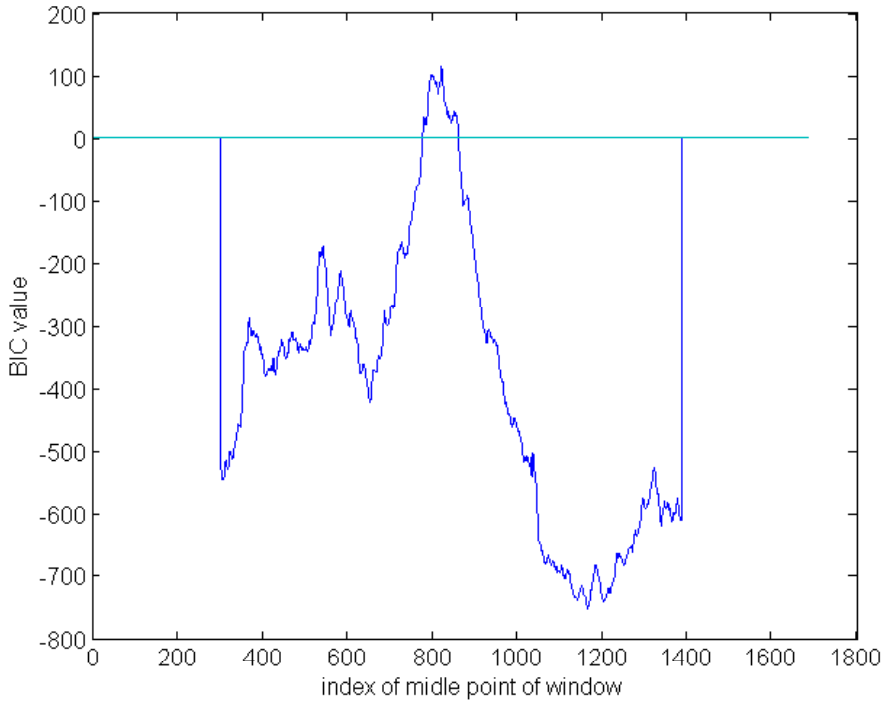
$$\mu_2 = \frac{1}{N-b}\sum_{i=N-b}^{N} x_i \tag{4.5}$$

$$\Sigma_2 = \frac{1}{N-b}\sum_{i=N-b}^{N}(x_i - \mu_2)(x_i - \mu_2)' \tag{4.6}$$

then we can update Σ_1 and Σ_2 for new point b increased by 1, to do this we have to remove scale. It means multiply it by b or $N-b$ respectively. Then to add a new scale, divide it by $b+1$ or $N-b-1$

$$\Sigma_1 new = (\Sigma_1 * b + (x_b - \mu_1) * (x_b - \mu_1)')/(b+1) \tag{4.7}$$

Figure 4.3: ΔBIC for sliding window.



On this picture are ΔBIC values computed by a sliding window.

$$\Sigma_2 new = (\Sigma_2 * (N - b) + (x_b - \mu_2) * (x_b - \mu_2)')/(N - b - 1) \tag{4.8}$$

where μ_1 and μ_2 are staying the same. This method is very noisy so I have decided to use this algorithm only for restricted number of steps and then recompute covariance matrices Σ_1, Σ_2 and their means μ_1, μ_2. This method is very fast a large number of steps before recomputing is chosen but is less reliable. For small number of steps, it is slower but more reliable. After some tests I have chosen a 100 steps to recompute everything.

4.4.2 Algorithm Description

How it is designed in steps:
1. take window from start, length 4 seconds and set b to the index given by start and offset
2. compute Σ, Σ_1, Σ_2 and μ, μ_1, μ_2
3. compute $\Delta BIC(b)$
4. increase b by 1
5. if we are at the end of window, go to 8.
6. if Σ_1 and Σ_2 not recomputed for given maximal number of steps, go to 2
7. update matrices as described in 4.7 and 4.8 and go to 3
8. if any $\Delta BIC(b)$ from current window is > 0 we have found a changing point on position b where $\Delta BIC(b)$ is maximal, set start index to this position and go to 1 if there are still at least 4 s to the end of audio stream.
9. if no changing point found add another 4 seconds to window and repeat the algorithm if there are still enough data to the end of audio stream.

This algorithm tends to be much faster because I have avoided 99% computations of covariance matrices.

4.4.3 Variants of Optimized Algorithm

In this work some other variants of optimized algorithm were implemented. The change is in the means used in equations 4.7 and 4.8. For shorter description I will call this optimized algorithms as optim1 .. optim4. Algorithm described above is optim1.

optim2 algorithm recomputes means each time when tested point b is increased before updating matrices according to equations 4.7 and 4.8:

$$\mu_1 = \frac{1}{b}\sum_{i=1}^{b} x_i \tag{4.9}$$

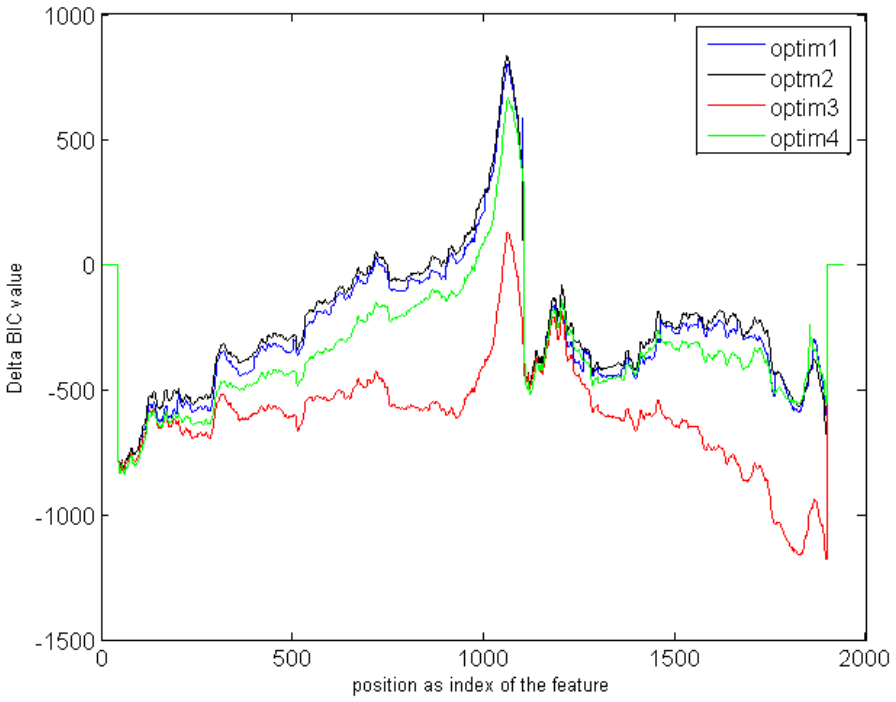
$$\mu_2 = \frac{1}{N-b}\sum_{i=N-b}^{N} x_i \tag{4.10}$$

optim3 lets the mean values μ_1 and μ_2 being the same as in the first estimation.

optim4 expects that the means μ_1 and μ_2 are the same and are equal to the mean of the whole window μ.

Lets see how the results looks like for a file with changing point at 1050.

Figure 4.4: Four Methods of Optimized Segmentation via BIC.



All the method have made a significant peak around the right changing point. But it can be seen that the method optim3 tends to have lower values with the window length increasing. Other methods gave a good results.

4.5 Tests on Synthetic Data

All methods described above were tested on a set of files made as an random concatenations of speech taken from TIMIT database. Each test file was a concatenation of 10 s speech 2 different speakers from TIMIT database. Silence was removed before parametrization of speech using my algorithm working with zero crossing rate and threshold. The MFCC vectors with 12 coefficients and 12 delta coefficients were used.
In the tables below we can see test results. If changing point was found in region +-0.5 s around known changing point, it was taken as correct. If it was found outside this region it was counted as false alarm and if there wasn't found any changing point it was counted as a miss. Tests were made for 4 types of concatenation based on combinations of speakers gender described as (mm, ff, fm and mf) where 'm' like male and 'f' like female. The last column average mismatch is saying how many frames were the correct answers from the expected changing point (one frame is 10ms).

method	type of concatenation	correct	false alarm	missed	count of test files	avg miss
optim1	mm	97.10	10.81	2.90	1036	8.29
optim1	ff	89.74	28.83	10.26	1540	11.31
optim1	fm	98.56	16.84	1.44	1045	6.29
optim1	mf	99.14	9.57	0.86	1045	3.99
optim2	mm	97.49	6.27	2.51	1036	5.31
optim2	ff	89.94	20.39	10.06	1540	9.06
optim2	fm	99.43	10.43	0.57	1045	3.43
optim2	mf	99.33	6.60	0.67	1045	2.38
optim3	mm	40.06	3.28	59.94	1036	5.45
optim3	ff	38.18	10.13	61.82	1540	9.62
optim3	fm	94.93	6.51	5.07	1045	6.05
optim3	mf	98.85	4.50	1.15	1045	3.08
optim4	mm	69.02	14.67	30.98	1036	7.15
optim4	ff	67.66	25.19	32.34	1540	12.30
optim4	fm	96.84	18.76	3.16	1045	6.65
optim4	mf	98.09	9.86	1.91	1045	4.25
BIC	mm	99.00	7.00	1.00	100	5.86
BIC	ff	93.14	19.61	6.86	102	7.88
BIC	fm	100.00	12.87	0.00	101	2.73
BIC	mf	98.02	11.88	1.98	101	2.31
T2BIC	mm	97.78	7.14	2.22	1036	5.00
T2BIC	ff	87.47	18.44	12.53	1540	8.24
T2BIC	fm	99.04	7.75	0.96	1045	4.05
T2BIC	mf	97.13	11.10	2.87	1045	3.24

Because optim2 algorithm is enough reliable and is about 10 times faster then T^2BIC algorithm. I have chosen to use it for segmentation step of my system.

Chapter 5

Clustering

Clustering is used to merge segments to groups (clusters) according to the speaker identity. I have decided to use the Bayesian Information Criterion for clustering. After the segmentation stage we have segments of speech and we expect that count of segments is much larger than count of speakers.

The main problem of clustering methods based on BIC is that there could be a lot of short segments which don't have enough data for a good covariance matrix estimation. In this work I am trying to make this problem less significant by skipping short segments in the first phase of clustering. So clustering is divided into 2 parts. In the first part the large segments are merged. I expect that each speaker has at least one large segment and therefore each speaker would have 1 cluster after the first phase of clustering. In the second phase the small segments are taken and each of them is added to the cluster with which it has best ΔBIC (minimal in this case). The idea of clustering decision is very similar to the segmentation decision. Lets expect two clusters c_i and c_j then we have 2 models M_1 and M_2. M_1 is described by mean and covariance matrix computed from all feature vectors of both c_i and c_j. M_2 is described by 2 means and covariance matrices each computed from feature vectors of one cluster. Then the equation for ΔBIC will be:

$$\Delta BIC(b) = \frac{1}{2}(Nlog|\Sigma| - N_i log|\Sigma_1| - N_j log|\Sigma_2|) - \frac{1}{2}\lambda(d + \frac{1}{2})d(d+1))logN \qquad (5.1)$$

Where Σ belongs to M_1, Σ_1 and Σ_2 belongs to M_2. N_i is the length of cluster c_i, N_j the length of cluster c_j and N is $N_i + N_j$. dim is dimension of feature vector and λ is a penalty factor. If ΔBIC is less than zero the clusters should be merged otherwise they should remain splitted.

The first stage of clustering consists of a loop in which we are computing ΔBIC for each pair of clusters c_i and c_j. If ΔBIC for some pair of clusters is less then 0 we merge the clusters c_i and c_j where ΔBIC is minimal. If ΔBIC is larger than zero for all pairs of clusters we stop the loop and the first stage of clustering is finished. In the beginning of this stage each cluster contained 1 large segment. At the end we should have clusters corresponding to the speakers identities. The count of clusters should be same as the count of speakers in the meeting record and each cluster contain all large segments of the corresponding speaker.

In the second stage we need to add small segments to the clusters. Estimation of covariance matrices for small segments could be noisy but usually the ΔBIC value is minimal for appropriate cluster. In this stage we don't care if ΔBIC is larger then zero because each small segment should be added to the corresponding cluster. It's better not to

recompute covariance matrices of clusters after adding small segments. Small segments have often badly estimated covariance matrices. It could lead to adding some small segments to the bad clusters. It causes the recomputed covariance matrix beeing less reliable than the original one very often. The original cluster covariance matrix is usualy enough reliable for adding small segments and avoiding recomputation leads to faster algorithm.

There are two main reasons for skipping small segments in the first phase of segmentation. The first was mentioned above, poor covariance matrix estimation could lead to adding small segments to the bad cluster. This would inflict worse estimation of cluster. The second reason is even more important. The small segments could have ΔBIC value larger then zero for all other clusters and it will inflict a creation of new cluster which doesn't belong to any speaker. Skipping of small cluster can be done because we expect that each speaker has at least 1 long speech. This expectation was right in all meeting records I were working with.

I was also experimenting with simpler and faster clustering algorithm. This method differs from the first one int the first stage where the large segments are processed. The large segments are used to build cluster where one segment is considered one cluster in the beginning. Further we take the largest cluster and compute ΔBIC with each other cluster. All clusters having ΔBIC value less than zero are added to this cluster. Clusters merged in this step are removed and we repeat the process of merging for the largest remaining cluster. This is repeated until there are some clusters remaining. Because this method is not choosing the best ΔBIC it depends a lot on the value of λ in ΔBIC computation. It can make more incorrect decisions than the previous method. It's why I have chosen a combination with the first method. In first stage we are merging pairs clusters with negative ΔBIC immediately but we are using smaller λ. This will lead to less merging but it can be considered more reliable. In the second stage we use bigger λ and we use algorithm of finding best pairs as it was described above.

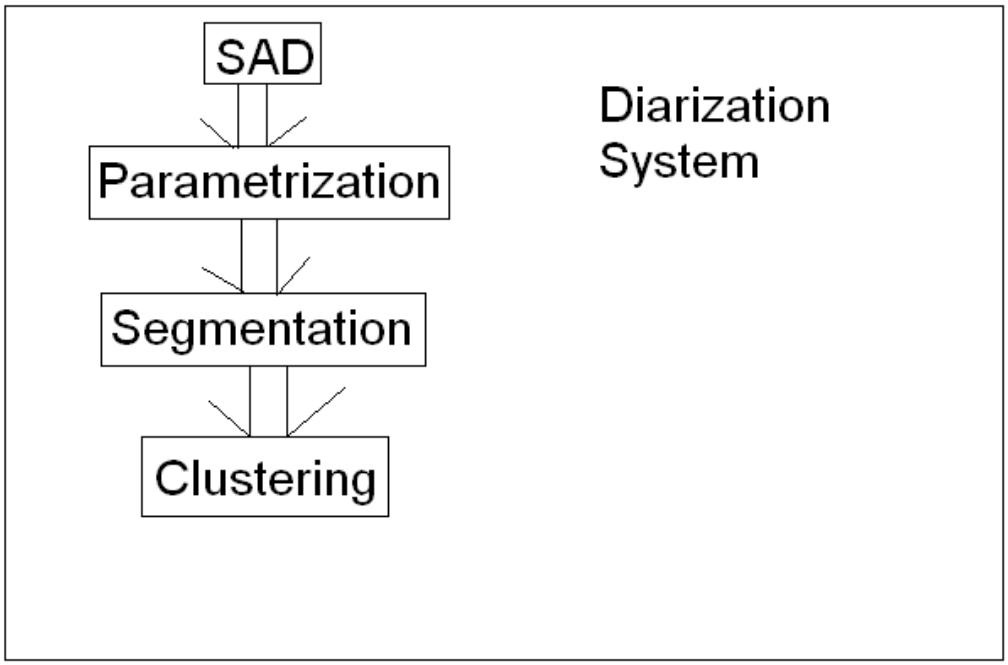
The method proposed first is slower but is more reliable so it will be used in the system.

Chapter 6

System Description

This chapter is describing the system realization. System consists of several stages as can be seen on the next picture 6. Theory and algorithms for most of them was described earlier. Here is described how the algorithm were used in system implementation.

Figure 6.1: Diarization System Stages.



6.1 Speech Activity Detection

In first stages of development of this system I was working with high quality data. So simple thresholding based on Zero Crossing Rate (ZCR) algorithm was enough reliable to discriminate between speech and silence. But it is not the best for diarization system

because you have to expect some noise and interrupts in the records. The method based on ZCR with threshold was working very bad for real meetings records. The records of meetings were recorded by multiple distant microphones and it leads to the different loudness of speech of different speakers. Because each meeting room has another recording equipment it would be of necessity to modify threshold in ZCR for each meeting room. This reasons made me to use some more sophisticated method for Speech Activity Detection. I have chosen to use segmentation system based on Hidden Markov Models developed by Jan Hovorka [4] on our faculty. This system was developed for detecting speech and music in audio records, it is also able to detect speech in the noisy record so it can be used for my experiments.

6.2 Parametrization

I have chosen to use mel frequency cepstral coefficients. Feature vector consists of 12 coefficients and 12 delta coefficients, each vector has 24 elements. Parametrization is computed in a 25 ms long window each 10 ms. For estimation of feature vector was used program HCopy, part of the HTK Hidden Markov Model Toolkit [9]. HCopy stores parameters into the file in the htk format. It means binary file with header defined in HTK. Because my system is implemented to use HTK header the change if parameters can be done easily.

6.3 Segmentation

Theory of segmentation was described in it's own chapter 4. Even the choice of algorithm and his function was described there. Implementation of algorithm in the system differs only a little from the optim2 algorithm described in the chapter Segmentation. Oonly change is that it uses information from Speech Activity Detection as the borders of segments and segmentation by optim2 algorithm is done only for larger blocks of speech (>5 s).The process could be described in four steps: 1. Load information from SAD (typically load a label file).
2. Create segments according to labels (borders of speech are used as borders of segments).
3. For all segments longer then 4s try segmentation.
4. If some change points are found, split the segment.

6.4 Clustering

Basics of clustering algorithm was described in chapter 5. Here will be given more accurate description of concrete algorithm used in my system. First we take large segments (>5 s) and make cluster from them. Each large segment corresponds to 1 cluster in the beginning of clustering. Then we need to compute covariance matrices for each cluster. Then we compute ΔBIC for each pair of clusters and store it into table. Now we need the main loop. In this loop 2 most similar clusters i,j are merged into new cluster taking index i. Cluster j is not used in the next iterations. We need to recompute covariance matrix of new cluster i and also update the table of ΔBIC values. The process is stopped if there is only 1 cluster or if all ΔBIC values are greater than zero. Now we need to place small segments into the corresponding clusters. It's done in the second loop which is adding each small segment into the cluster with minimal ΔBIC between them.

1. Create clusters - 1 large segment = 1 cluster
2. Compute covariance matrices of all vectors
3. Compute ΔBIC for each pair of clusters and store it into table
4. If any ΔBIC in the table is less than zero continue otherwise go to step 9
5. Merge clusters i,j with minimal ΔBIC into cluster i
6. Recompute covariance matrix for cluster i
7. Remove j from table of ΔBIC values and update records for i
8. Go to step 4. 9. For each small segment compute ΔBIC for this segment and each cluster 10. Add segment to the cluster with minimal ΔBIC

Chapter 7

Test

7.1 Test description

The NIST RT05 [7] meeting data was used for testing. It consists of records of meeting recorder in several meeting rooms using multiple distant microphones. In each test file up to 8 speakers are speaking. Records are of different quality because of different recording equipment was used. Meeting records are about 1/2 - 1 hour long. Tests were done on 10 meeting records with total length about 6 hours.

Evaluation of results is done using equation for Speaker Diarization Error (SDE):

$$\frac{\int C(wrongspeaker)dt}{\int C(anyspeaker)dt}$$

Which can be described as time of incorrectly clustered segments divided by the time of speech in meeting.

For evaluation was used script provided by NIST [7]. Speech music discrimination system [4] was used to detect speech .

7.2 Test results

Results using speech-music discrimination system as SAD:

1215.75	Total time in a state to be segmented (in seconds)
546.50	Time in a correctly segmented state (in seconds)
669.25	Time in an incorrectly segmented state (in seconds)
0.55	SEGMENTATION ERROR

Result is not very good. It is beacause of imperfect Speech Activity detection and because my clustering algorithm is set to prefer more clusters then speakers if this clusters have good purity. Error can't be lower than 30% which is aproximately the error of SAD.

Chapter 8

Conclusion

In this work I was trying to propose diarization system which would offer better results than systems commonly used nowadays. Proposed system is partially applicable for diarization task but is not very reliable. So I wasn't very successful. System is unsupervised and if not-counting Speech Activity Detection it works without any training. In future the system can be improved by using better SAD. Another great improvement could be adding agglomerative resegmentation. It would be best to combine it only with the first part of clustering (the part dealing with large segments). Another possible improvement could be using other feature vector I haven't done any large testing of different sets of features so I can't declare that the selected vector is the best.

Bibliography

[1] S. Chen and P. Gopalakrishnan. *Speaker, environment and channel change detection and clustering via the Bayesian information criterion.* in Proc. DARPA Broadcast News Transcription Understanding Workshop, 1998. pp. 127-132.

[2] S.B. Davies and P. Mermelstein. *Comparison of Parametric Representations for Monosyllabic Word Recognition in Continuously Spoken Sentences.* IEEE Trans. Acousticsm Speech and Signal Processing, vol. ASSP-28, no. 4, pp. 357-366, 1980.

[3] A. Haubold and J. R. Kender. *Accommodating Sample Size Effect on Similarity Measures in Speaker Clustering.* Department of Computer Science, Columbia University, 2006.

[4] J. Hovorka. *Audio Data Segmentation.* Department of Computer Graphics and Multimedia FIT BUT, 2006. Master Thesis.

[5] D. van Leeuwen. *The TNO speaker diarization system system for NIST RT05s for meeting data.* NIST 2005 Spring Rich Transcrition Evaluation Workshop, 2005.

[6] D. A. Reynolds and P. Torres-Carrasquillo. *The MIT Lincoln Laboratory RT-04F Diarization Systems: Applications to Broadcast Audio and Telephone Conversations.* DARPA EARS RT-04F Workshop, White Plains, NY, Nov. 8-11, 2004.

[7] WWW stránky. National institute of standards and technology, speech group. `http://www.nist.gov/speech/`.

[8] `Wei-Ho Tsai, Shih-Sian Cheng, and Hsin-Min Wang. *Speaker clustering of speech utterances using a voice characteristic reference space.* INTERSPEECH-2004, 2937-2940, 2004.`

[9] `WWW. Htk speech recognition toolkit. http://htk.eng.cam.ac.uk/.`

[10] `B. Zhou and J. Hansen. *Efficient Audio Stream Segmentation via the combined` T^2 `Statistic and Bayesian Information Criterion.* IEEE Transactions on Speech and Audio Processing, 2005.`