# Evolutive Speaker Segmentation using a Repository System

*Xavier Anguera, Javier Hernando*

Department of Signal Theory and Communications, TALP Research Center
Technical University of Catalonia (UPC), Barcelona, Spain
{xanguera, javier}@gps.tsc.upc.es

## Abstract

When performing blind speaker segmentation one of the main problems is not knowing how many speakers appear in a conversation and wether they appear once or more than once. In this paper, an iterative method, which is based on the Evolutive-HMM is presented. Two main improvements to this system are introduced. On one hand, a repository generic speaker is used to model all utterances and all speaker models are derived from this iteratively. Different normalization of the scores are applied to the repository and the speakers to emphasize speaker changes. On the other hand, in all cases we use Gaussian Mixture Models (GMM) for their flexibility compared to an HMM structure.
This method has been successfully tested using multi-speaker speech sequences generated by concatenation of speech segments from Speecon.

## 1. Introduction

Blind speaker segmentation consists on finding the start and end points where there is speech from only one person in a speech sequence. The system initially does not have any information about the number of speakers or their identity. This is mainly used in speaker indexing, where the different segments are found in different sessions and then some are clustered together according to acoustic similarities. It is also used in speaker recognition, where the separation of the different speakers in a speech sequence is previous to the speaker recognition algorithm.

In the literature we can find various methods for addressing speaker segmentation. Metric-based techniques [1] define acoustic distance measures that evaluate the similarity between two adjacent windows. By scrolling such windows a distance curve can be computed and peaks are defined as speaker changing points. On the other hand, the Bayesian Information Criterium (BIC) [2] is a widely adopted method because of its robustness and for being threshold free. The speaker changing point is searched within a window, increasing its size progressively until it is detected. Another method is the Evolutive-HMM (E-HMM), that was first presented by [3], [4]. Speaker changes are modelled as state changes in a HMM, modelling one speaker in each state. New states are added until a maximum score is reached.

The architecture of the proposed segmentation system is based on E-HMM. It iterates on the number of speakers and on the segmentation limits between speakers. A Repository model is initially created to model all the observation vectors. New speaker models are created by adapting a Universal Background Model (UBM) with observation vectors subtracted from the Repository. This process takes place while there is enough observations in the repository to adapt new speakers and the likelihood of the system increases. By doing this we avoid local speakers fluctuations by clearly modelling the non assigned regions.

Gaussian Mixture Models (GMM) are used to model each speaker and the repository model. The score curves from the repository and the speakers, calculated by the GMM models from the input signal, are normalized differently. Such normalization emphasizes the segments remaining in the repository from the already subtracted regions. The segmentation limits are defined by a weighted comparison of the score curves. This is used to better emphasize the speaker segments. We call this proposed method Repository-based Evolutive-GMM (RE-GMM).

In section 2.1 we will overview the Evolutive-HMM system. In 2.2 we present the new evolutive-GMM system architecture and in 2.3, 2.4 and 2.5 we will further explain the concept we introduce. In section 4 the experimental setup is presented and results are explained. Finally conclusions are stated in section 5.

## 2. Segmentation Model

### 2.1. Evolutive-HMM System

The architecture of our proposed RE-GMM closely relates to the E-HMM approach [3],[4]. The system starts by adapting a first speaker from a Universal Background Model (UBM) with all the observation vectors from the input signal. This model is used to create a score curve by decoding the input signal. The best 3 seconds are used to adapt a second speaker model from the UBM model and create a 2 states HMM together with the first speaker. The first speaker is also adapted from the UBM with the remaining observation vectors. Then the HMM is used to decode the input signal and a segmentation of the observation vectors is found by looking at which state is the signal at each instant. The speakers are adapted from the UBM model with the resulting data. These intraspeaker iterations take place until the total decoding likelihood decreases. Upon exiting the intraspeaker loop, a speaker stop criterium is assessed to see wether the newly created speaker should be kept. If it succeeds a new speaker is added to the HMM using the same procedure. When it does not, the segmentation prior to this last speaker becomes the result of the process.

### 2.2. Repository E-GMM System Architecture

Let us define the signal to be segmented as consisting of a set of observation vectors, $\mathcal{O} = \{o_1, o_2, \ldots, o_T\}$ created by analysis of the input speech signal.

A group of M sound classes (i.e. speakers) is defined, which best models a region of the group of observation vectors $\mathcal{O}$, which we will reference as $\mathcal{S} = \{s_m / m \in \{1 \ldots M\}\}$. Such

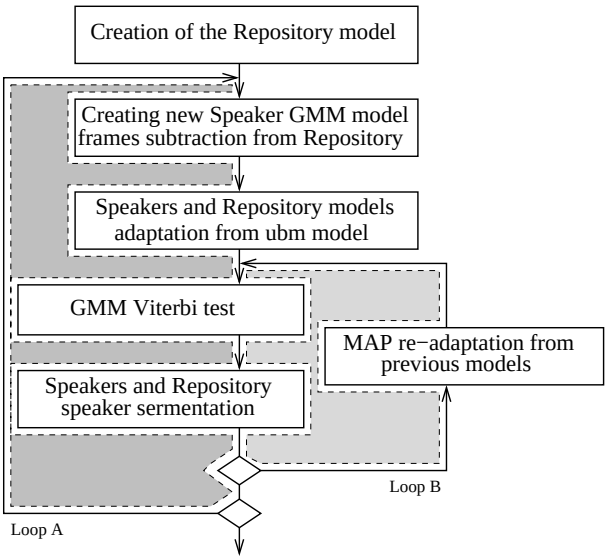


Figure 1: *Speaker segmentation system flow diagram.*

regions can consist of a contiguous segment of the signal or more than one.

Also, a repository class $\mathcal{R}$ is defined which includes all observation vectors not in $\mathcal{S}$, covering all the remaining region ($\mathcal{O} = \mathcal{R} \cup \mathcal{S}$).

Both speakers and repository are modelled by one single N dimensional GMM with diagonal covariance matrix.

The Repository model is initially adapted from a UBM model with all the observations in $\mathcal{O}$. Then the first speaker is also adapted from a UBM using the 3 seconds of biggest likelihood from decoding the input signal with the repository model. Each time a new speaker is created (loop A in figure 1), all models are adapted from the UBM model with the observations assigned to it from the previous segmentation. Within a speaker, at each iteration the models are always adapted from the previous ones (loop B in figure 1) with the current segmentation of $\mathcal{O}$ Two stopping criteria are assessed at each iteration at the end of loops A and B. Both consist on the percentage of variation between the total accumulated likelihood between the previous and current iterations. It is set to 10% for loop A and 1% for B. For loop A (inter-speakers loop) we also check that the repository is modeled by a minimum number of observation vectors.

As with the E-HMM the prior step segmentation is selected as resulting speakers segmentation when the process finishes. The residual observation vectors belonging to the repository model are split in equal parts between the two models next to them.

### 2.3. Models Adaptation

When initializing the system, the subset $\mathcal{R}$ of $\mathcal{O}$ which we call repository, models all observation vectors $\mathcal{O}$ (amount of speakers M=0) and becomes smaller as M increases. The repository should tend to $\mathcal{R} = \oslash$, which would happen for a value of $M = M_{opt}$. In fact, R maintains a residual amount of frames due to portions of the signal not well modelled by any speaker. Alternative criterions have to be set to stop increasing M.

In the RE-GMM model, both classes $\mathcal{R}$ and $\mathcal{S}$ are derived from a UBM by Maximum a Posteriory (MAP)[6] adaptation of the mean vectors in the following way:

$$\mu_g(n) = \alpha\mu_g(n-1) + (1-\alpha)\mu_g^{ML}(n) \qquad (1)$$

Where $\mu_g(n)$ stands for the individual means of each of the Gaussian in each of the models at iteration n, and $\mu_g^{ML}(n)$ is the EM-ML estimate of the mean for that Gaussian given the training data. The factor $\alpha$ must be positive and weights the importance of the previous value versus the EM-ML estimate in the adaptation.

MAP is used to adapt the models from loop B (figure 1) on each iteration. This is done to increase the difference between the modelled observations and the rest, allowing similar observations (from the same speaker, but outside of the acoustic class) to be selected by the next segmentation. To illustrate this, in figure 2, a certain model is initially adapted from the UBM with subset 1 from the speech signal, and then adapted with subset 2. The likelihood curves are shown for both iterations. The common observation vectors between subset 1 and 2 have the highest score after iteration 2. Subsets only adapted once (scores a1 and a3) result in lower scores and will stay approximately the same if they are not used any mode for adaptation. Special attention must be payed in selecting the $\alpha$ parameter so

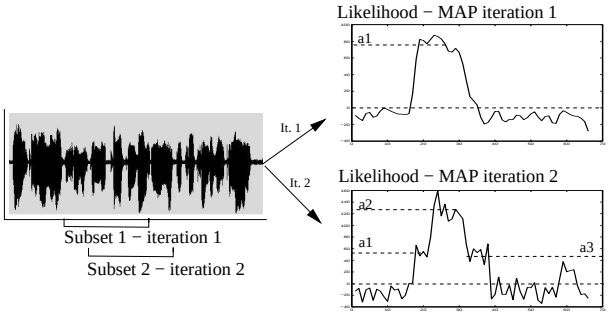


Figure 2: *Effect of adaptation in the score curves.*

that the models do not get overadapted with the adaptation data. In iteration 2 there are some observation vectors that increased their score and would probably be considered in the next iteration. A value of $\alpha = 0.7$ is used in our system.

By observing how adaptation affects the resulting scores and the models we have, we can define 2 different kinds of adaptation:

- The speaker models are created with a small amount of data and by adapting we want to expand the high likelihood region to cover the modelled speaker region, therefore we can call it *additive adaptation*.
- In the repository model, by adapting we want to iteratively reduce the areas of high likelihood, which should be assigned to the speaker models. Therefore we can call it *subtractive adaptation*.

### 2.4. Likelihood Normalization

When adapting the repository model, the regions which do not belong to it anymore but keep a constant score aren't desirable as they can still be selected in the segmentation step. No such problem is presented in the speaker models $s_i$ as they keep adding observation vectors.

We can eliminate the residual regions from the repository scores curve by normalization by it is previous iteration score curve. This way the region used for adaptation is emphasized

in the scores used for comparison, converting the subtractive scores into additive scores:

$$logL_n(X, s_c) = logP_n(X/s_c) - logP_{n-1}(X/s_c) \quad (2)$$

Where $L_n(X, s_c)$ indicates the normalized likelihood given the observation data and the speaker model $s_c$, and $P_n$ is the non normalized probability.

The speaker models are normalized by the UBM scores curve, commonly used to reduce the effect of the channel in the scores:

$$logL_n(X, r) = logP_n(X/r) - logP(X/ubm) \quad (3)$$

The effect of the repository normalization can be observed in figure 3, where the 2 different normalization procedures are represented. In the speaker model the normalized score signal slowly expands to model the speaker region. The repository model scores become more clear after normalizing, useful for the segmentation step.

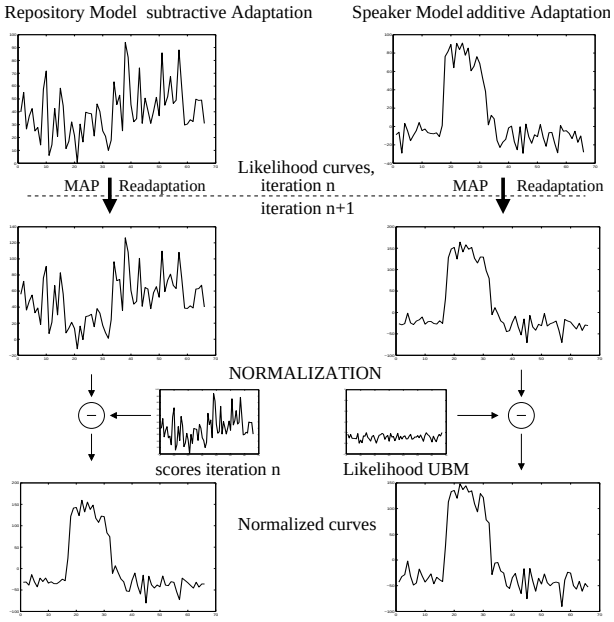


Figure 3: *Used normalization in the system models.*

As the amount of iterations increase, fewer observation vectors are assigned to the repository model. The normalized score curve for the repository tends to 0 as $M \rightarrow M_{opt}$.

### 2.5. GMM Model and Segments Evaluation

Each iteration within a speaker (loop A in figure 1) starts with a MAP adaptation of all the GMM models (Repository and speaker models) with the current segmentation information. Then the normalized likelihood of the $\mathcal{O}$ observation vectors set is computed for each GMM.

Using individual GMM to model the speakers and repository models presents a simpler structure and are more flexible to implement segmentation algorithms than HMM. They also accept different score normalizations to be applied to each of the model score curves. In the system presented, speaker changes are computed by a weighted comparison of all the likelihood curves averaged over a window of length T samples.

Upon decoding the input signal by each of the models, we obtain a score curve with a score value for each frame. Such curve is very noisy and therefore it is averaged in blocks of T samples, obtaining the score curve to be used in the segmentation step.

Considering M+1 models (the speakers and repository), after the calculation of the likelihoods and normalization, we obtain M+1 score curves ($\mathcal{L}_i/i = 1 \ldots M+1$).

Starting at observation t and duration T samples, and with weighting factor $\gamma$, the selected model that best fits that observations block is found by:

$$sel\_model(k) =$$

$$\max_j \begin{cases} \gamma \sum_{t=kT}^{kT+T} L_j & if j = sel\_model(k-1) \\ \frac{(1-\gamma)}{(N-1)} \sum_{t=kT}^{kT+T} L_j & otherwise \end{cases} \quad (4)$$

When taking the decision of point $sel\_model(k)$ it takes into account the previously selected model. In this way it resembles the HMM models $a_{ij}$ probabilities but averaging the scores over a block of T frame scores. The value T is equal to the resolution of the segmentation.

## 3. Experiments

### 3.1. Database

The proposed method was tested using an N speaker database artificially generated from SPEECON [6] clean sentences. This Database consists of 600 people recordings of varying length and contents. For this test the Spanish database was used. 90 speakers were taken and only 30 sentences from each speaker were selected, in which the speaker reads a sentence from 2 to 8 seconds long. Head microphone channel was used. From each set of 30 sentences from each speaker, 20 sequences were chosen to create the test sequences and 10 sequences were used to train the World Model. From the selected set for testing, 450 speech sequences were created consisting on 3, 4 and 5 speakers in equal parts. No sequences are repeated in any of the generates sequences. Within a sequence, all the speakers are different.

### 3.2. Speaker Segmentation System

The proposed system has been implemented using the Cambridge HTK toolkit.

Before parametrization, a voice activity detector (VAD) was applied to extract all the silence parts from the signal. These parts were discarded and are neither used in the test nor in the performance evaluation.

In the parametrization of the speech sequences, one observation vector was generated each 10ms. A 24 filter bank was computed and 16 MFCC static parameters and first order derivatives were extracted. Cepstral Mean Substraction (CMS) was applied to the result to reduce channel effects.

MAP adaptation was applied with $\alpha$ = 0,7 for all speaker models and repository model. In comparing the scores, a weighting of $\gamma$ = 0,6 was applied to the previously selected speaker model.

Measures were computed using 3 possible values of T ( T= $\{10, 20, 30\}$) to see the effect of modifying the average window length related to the influence of the segmentation resolution in the final results.

### 3.3. Evaluation and Results

In evaluating the system we use a speaker segmentation scoring measure as used in NIST Rich Transcription Evaluation [9].

The script used by NIST allows the reference and hypothesis speaker segments to have different labels. It calculates the segmentation percentage error for the optimum one-to-one mapping of reference speaker IDs to system output speaker IDs. We calculate such error for a collar value of 0.25 seconds, applied around each segment, and with no collar.

The speaker segmentation error scores using RE-GMM are presented in table 1. In order to evaluate the use of a repository model, the scores have been also calculated using the system without the repository model. This test system is referenced as E-GMM as it is similar to the E-HMM but using GMM models instead of HMM . All other parameters are kept the same in both systems. All values of T within the same method achieve

Table 1: *SPEECON M-speakers segmentation score (%) by block size.*

| Method | collar length | 10 | 20 | 30 |
|---|---|---|---|---|
| E-GMM | - | 20.68 | 19.50 | 19.07 |
| RE-GMM | - | 16.07 | 15.61 | 15.84 |
| E-GMM | 0.25 | 16.76 | 15.57 | 15.10 |
| RE-GMM | 0.25 | 13.15 | **12.73** | 12.9 |

very similar results which indicates the robustness of the methods to the value of T. The best case for RE-GMM is T = 20 frames, showing that it is a good tradeoff between resolution and average window size (which is 0.2 seconds when the frames sampling period is 10ms).

The percentage error has also been computed for the case where all the signal is assumed to be from one speaker. We obtained a score of 66.76% using a collar and 67.91% without collar.

In figure 4 the four scores for case T=20 are drawn. There is a 19.9% relative improvement between the two methods when using a collar, and a 18.2% when not using any collar. This shows how RE-GMM represents an important improvement to the present technique.

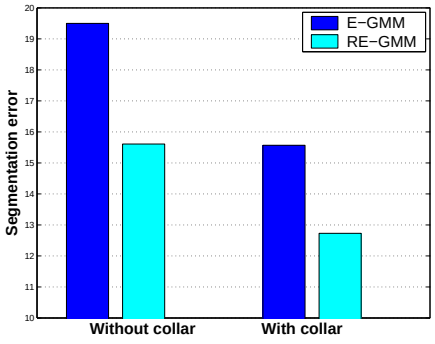


Figure 4: *Comparison between RE-GMM and E-GMM for T=20.*

## 4. Conclusions

In this paper we present a segmentation system which is a modification of the E-HMM system by using GMM models and a repository system. The repository initially models all the input signal and observation vectors are iteratively subtracted as new speakers are introduced. The effect of MAP adaptation over the models is explained and normalization is used to emphasize the difference among speakers and repository.

Our experiments with a database created from Speecon samples give very promising results. The scores were compared with the same system without using the repository model, improvement around 20% was achieved with the database used.

For all cases the segmentation scores are similar to the ones from the best systems in the NIST Speaker segmentation tasks, although here a different database was used.

## 5. Acknowledgements

Special thanks go to the TALP research group for their help and advise to develop this work. In particular, thanks to Climent Nadeu, Pablo Daniel Agüero, Jan Anguita, Mireia Farrús and Jordi Adell for their support.

## 6. References

[1] Hung, J.W., Wang, H.M., Lee, L.S. ”Automatic Metric-Based Speech Segmentation for Broadcast News Via Principal Component Analysis”. Proceedings of ICSLP’2000.

[2] Shaobing Chen, S., Gopalakrishnan, P.S., ”Speaker, Environment and Channel Change Detection and Clustering Via the Bayesian Information Criterion”, Proceedings DARPA Speech Recognition Workshop, 1998.

[3] Meignier, S., Bonastre, J.F., Igounet, S., ”E-HMM Approach for Learning and Adapting Sound Models for Speaker Indexing”, 2001: A Speaker Odissey, pp. 175-180, Chania, Crete, June 2001.

[4] Meignier, S., Bonastre, J.F., Fredouille, C., Merlin, T., ”Evolutive HMM for Multi-Speaker Tracking System”, ICASSP, June 2000.

[5] Moraru, D., Meignier, S., Besacier, L., Bonastre, J.F., Magrin-Chagnolleau, I., ”The ELISA Consortium Approaches in Speaker Segmentation During the NIST 2002 Speaker Recognition Evaluation”, ICCASP’03, Hong Kong.

[6] Siemund, R., Hoge, H., Kunzmann, S., Marasek, K., ”SPEECON - Speech Data for Consumer Devices”, Proceedings of the Second International Conference on Language Resources and Evaluation, vol II, pp. 883-886.

[7] Reynolds, D.A., ”Comparison of Background Normalization Methods for Text-Independent Speaker Verification”, Proceedings of the European Conference on Speech Communication and Technology, pp. 963-967, September 1997.

[8] Gauvain, J.-L., Chin-Hui Lee,”Maximum a Posteriori Estimation for Multivariate Gaussian Mixture Observations of Markov Chain”, in Transactions on Speech and Audio Processing, pp 291-298, Apr 1994.

[9] The NIST RT-03 Rich Transcription Evaluation plan, http://www.nist.gov/speech/tests/rt/rt2003/fall/docs/rt03-fall-eval-plan-v9.pdf