Automatic Cluster Complexity and Quantity Selection: Towards Robust Speaker Diarization

Xavier Anguera[1,2], Chuck Wooters[1], and Javier Hernando[2]

[1] International Computer Science Institute, Berkeley CA 94704, USA,
[2] Technical University of Catalonia, Barcelona, Spain
{xanguera, wooters}@icsi.berkeley.edu

Abstract. The goal of speaker diarization is to determine where each participant speaks in a recording. One of the most commonly used technique is agglomerative clustering, where some number of initial models are grouped into the number of present speakers. The choice of complexity, topology, and the number of initial models is vital to the final outcome of the clustering algorithm. In prior systems, these parameters were directly assigned based on development data, and were the same for all recordings. In this paper we present three techniques to select the parameters individually for each case, obtaining a system that is more robust to changes in the data. Although the choice of these values depends on tunable parameters, they are less sensitive to changes in the acoustic data and to how the algorithm distributes data among the different clusters. We show that by using the three techniques, we achieve an improvement up to 8% relative in the development set and 19% relative in the test set over prior systems.

1 Introduction

The goal of speaker diarization is to segment an audio recording into speaker-homogeneous regions [1]. Typically, this segmentation must be performed with little knowledge of the characteristics of the audio or of the participants in the recording. For example, we may know the source and date of the audio recording (e.g. CNN Nightly News or a NIST meeting), but we typically do not know how many speakers occur in the recording, how many speakers are male vs. female, whether there are commercials, music, or other noises, etc.

Typically, most speaker diarization systems use algorithms that are governed by tunable low-level parameters that are adjusted using development data of the same sort as the testing data. This is the case, for example, for the acoustic models used, the penalty factor used on the Bayesian Information Criterion (BIC) to compare models [2], and some initial parameter values such as the number of initial speaker clusters, the number of Gaussian mixtures per model at each state of the process, and the average speaker turn length. Such systems perform poorly when conditions change between the train and test sets; also selecting a constant value for all the recordings in a set lead to the omission of some particularities of each recording, resulting in a suboptimal result.

In this paper, we present three algorithms that help determine important parameters in the clusters modeling, namely the number of Gaussian mixtures per model at each step of the processing, the number of initial models in the system, and the topology of each acoustic model. In order to determine the number of Gaussian mixtures and the number of initial clusters, the algorithms base their selection on information on each particular recording rather than defining a pre-fixed value for all recordings of a certain type. In order to do this, we define a parameter that we call the Cluster Complexity Ratio (CCR), which defines a ratio between the data being modeled and the mixtures needed to represent it. The CCR ratio is defined using development data, and it is used to define recording-specific values for the above mentioned parameters.

The third novelty presented in this paper is the elimination of the dependency of the acoustic models on the average speaker turn length. This is achieved by modifying the acoustic modeling topology by changing the probabilities of self-loop and transition in the last state. By doing so, we can apply a minimum duration for a speaker turn while not influencing the final duration. While setting a minimum duration for speaker turns is advantageous for the processing of the recordings and can be set to be independent of the kind of recording we encounter, the average speaker turn duration is quite variable between individual recordings and recording types. It is therefore interesting to let the acoustic data define when the speaker turn finishes once it achieves a minimum length.

In section 2, we present the speaker diarization algorithm with the proposed algorithms. In sections 3 through 5, we present the algorithms in detail. Then the experiments are presented, and finally conclusions are drawn from them.

2 Agglomerative Speaker Diarization System

As explained in [3] and [4], the speaker clustering system is based on an agglomerative clustering technique. It initially splits the data into K clusters (where K must be greater than the number of speakers and is chosen by the presented algorithm), and then iteratively merges the clusters (according to a merge metric based on ΔBIC) until a stopping criterion is met. Our clustering algorithm models the acoustic data using an ergodic hidden Markov model (HMM), where the initial number of states is equal to the initial number of clusters (K). Upon completion of the algorithm's execution, each remaining state is taken to represent a different speaker. Each state in the HMM model contains a set of MD sub-states, imposing a minimum duration on the model (we use $MD = 3$ seconds). Within the state, each one of the sub-states shares a probability density function (PDF) modeled via a Gaussian mixture model (GMM). A modification to this architecture presented in this paper avoids any maximum time duration constraints on the speaker turns, as further explained in section 5.

The following items show step by step the clustering algorithm used in the meetings domain, where we include the novel processing presented in this paper (explanation on previous systems can be found in [6] and [3]):

1. Run speech/non-speech detection on input data.

2. Extract acoustic features from the data and remove non-speech frames.
3. **(new)** Estimate the number of initial clusters K and set their initial model complexity (number of Gaussian mixtures per model).
4. Create models for the K initial clusters via linear initialization.
5. Perform several iterations of segmentation and training to stabilize the data among the different models.
6. **(new)** Adjust the complexity of each resulting model according to the data assigned to them and retrain all models.
7. Perform iterative merging using the following steps:
   (a) Run a Viterbi decode to resegment the data.
   (b) **(new)** Adjust the models complexity according to the newly assigned data.
   (c) Retrain the models using the Expectation-Maximization (EM) algorithm and the segmentation from step (a).
   (d) Select the cluster pair with the largest merge score (based on ΔBIC) that is > 0.0.
   (e) If no such pair of clusters is found, stop and output the current clustering.
   (f) Merge the pair of clusters found in step (c). The models for the individual clusters in the pair are replaced by a single, combined model.
   (g) Go to step (a).

For the merging and clustering stopping criteria, we use a variation of the commonly used Bayesian Information Criterion (BIC) [2]. The ΔBIC compares two possible models: two clusters belonging to the same speaker or to different speakers. The variation used was introduced by Ajmera et al. [4], [5], and consists of the elimination of the tunable parameter λ by ensuring that, for any given ΔBIC comparison, the difference between the number of free parameters in both models is zero.

Both the estimation of the initial number of clusters and the model complexity selection ensure that each individual show starts at an optimum number of clusters, and that each cluster is able to model well the data in it. Although theoretically the initial number of clusters should not be a decisive parameter for an agglomerative clustering system, in practice it turns to be an important factor in the performance of a system. This is probably due to the different resulting number of Gaussian mixtures that are used to model each cluster at each stage of the process. It is therefore important to determine a tradeoff between the number of Gaussian mixtures assigned to each cluster and the number of initial clusters. In sections 3 and 4, we present the relationship between both parameters through a newly defined parameter called the Cluster complexity Ratio (CCR).

3 Model Complexity Selection

The acoustic models used to represent each cluster are a key part of the agglomerative clustering process. On the one hand, comparing their likelihood given

the data is how we decide whether two models belong to the same cluster or not. On the other hand, they are used in the decoding process to redistribute the acoustic data into the different clusters on every iteration.

When designing their size, an important decision is whether we use fixed models (meaning a fixed number of Gaussian mixtures from start to finish), or if we allow the number of Gaussian mixtures to vary according to time or occupancy. Using fixed models is a viable alternative, but runs into the problem of having sufficient training data when the we set the number of Gaussian mixtures to be high, or being too general a model when it is set to be small.

Furthermore, when comparing two models via BIC, if they are too general they tend to over-merge, and when they are too specific to the data they under-merge. Therefore it is important to find a tradeoff on the number of mixtures used (model complexity). This has been addressed in our past systems ([6] and [3]) by using variable complexities as the merging process progresses. In such systems, all cluster models (regardless of their size) are initially trained using a fixed number of Gaussian mixtures. Upon merging any two clusters, the data from both original clusters are merged and a new cluster model is created as the sum of both parents' Gaussian mixtures. This is a variable complexity approach that changes over time.

Such an approach has a drawback that is addressed with our proposed technique. Even though when we start the algorithm, we have the same amount of data assigned to each individual cluster (due to using linear initialization of the available data into clusters), when iterating over decoding the data with the models and merging the different models we obtain clusters with much less data assigned to them that are still modeled with the same complexity than much more populated ones. When performing a BIC comparison, we are comparing more specific models to more general ones, suffering in system performance.

We present an algorithm that selects the number of mixtures to be used when modeling each cluster according to its occupancy. This could be referred to as an **occupancy driven approach**. After each change in the amount of data assigned to each cluster (due to a segmentation), we count the number of acoustic frames that are assigned to each of the models and determine the number of mixtures by:

$$M_i^j = round(\frac{N_i^j}{CCR_{gauss}}) \tag{1}$$

The number of Gaussian mixtures to model cluster i at iteration j (M_i^j) is determined by the number of frames belonging to that cluster at that time (N_i^j) divided by a constant value (CCR_{gauss}) that we call Cluster Complexity Ratio, fixed across all meetings.

In both approaches (time and complexity driven), the total number of mixtures used over all models remains constant in average, being distributed between the different cluster models as described above. This allows tracking of the system evolution by inspection of the Viterbi decoding total likelihood, which can be compared across merging iterations.

When the complexity of any given model changes, we update the model in one of two alternative ways: a) when the final complexity is greater than the initial one, we iteratively split the Gaussian with the biggest posterior probability into two (in the same way as performed within HTK) until the desired number is reached; or b) when the final complexity is smaller than the initial, we erase the initial model and obtain a new model of the given complexity by initializing it to the desired number of Gaussian mixtures given the data.

4 Automatic Selection of the Initial Number of Clusters

In order to perform an agglomerative clustering on the data we need to define an initial number of clusters. This value needs to be higher than the actual number of speakers to allow the system to perform some iterations before finding the optimum number. It also cannot be too big, as each model needs a minimum cluster occupancy to be trained properly, and to avoid unnecessary computation.

In prior work ([6] for the meetings domain and [3] for broadcast news data), the number of initial clusters was fixed within each domain that we work on. In the meetings domain, it was set to either 10 or 16 initial clusters, and in the broadcast news domain it was set to 40 initial clusters. The selection of these values has to be tuned to be greater than the possible number of speakers in any given recording while maximizing the performance.

With the following method, we can estimate the number of initial clusters on a per recording basis by taking into account the total amount of data available for clustering:

$$K = \frac{N_{total}}{G_{clus} CCR_{gauss}} \tag{2}$$

We make the number of initial clusters a function of the amount of data available for clustering, the number of Gaussian mixtures we want to assign per cluster (we use, as in prior work, $G_{clus} = 5$) and the Cluster Complexity Ratio CCR_{gauss}. This initializes the system using an average complexity of G_{clus} and the amount of data per cluster as defined by CCR_{gauss}, which is the same as when defining the models complexity during the previously presented algorithm. This technique does not try to guess the real number of speakers present in a recording, but rather sets an upper boundary to the number of speakers that is closely coupled with the complexity selection algorithm and which allows a correct modeling of each initial cluster for each particular recording.

5 Acoustic Modeling Without Time Restrictions

Our clustering algorithm models the acoustic data using an ergodic hidden Markov model (HMM), where state corresponds to one of the initial clusters. Upon completion of the algorithm's execution, each remaining state is considered to represent a different speaker. Each state contains a set of MD sub-states,

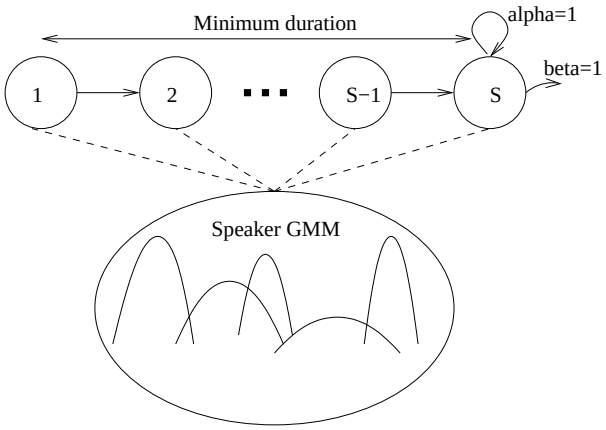

Fig. 1. *Cluster models with Minimum duration and modified probabilities*

as seen in figure 1, imposing a minimum duration of each model. Each one of the sub-states has a probability density function modeled via a Gaussian mixture model (GMM). The same GMM model is tied to all sub-states in any given state. Upon entering a state, at time n the model forces a jump to the following sub-state with probability 1.0 until the last sub-state is reached. In that sub-state, it can remain in the same sub-state with transition weight α, or jump to the first sub-state of another state with weight β/M, where M is the number of active states/clusters at that time. In prior publications, these were set to $\alpha = 0.9$ and $\beta = 0.1$ (summing to 1).

One disadvantage of using these settings is that it imposes an implicit duration model on the data beyond the minimum duration MD set as a parameter. Such duration modeling changes as we modify the MD value, as illustrated by equations 3 and 4.

$$lkld_{AA} = prob(x(0)|\Theta_A) \prod_{i=1}^{MD-1} (1 \cdot prob(x(i)|\Theta_A)) \cdot \prod_{i=MD}^{2 \cdot MD-1} (\alpha \cdot prob(x(i)|\Theta_A)) \tag{3}$$

$$lkld_{AB} = prob(x(0)|\Theta_A) \prod_{i=1}^{MD-1} (1 \cdot prob(x(i)|\Theta_A)) \cdot \frac{\beta}{M} prob(x(MD)|\Theta_B) \prod_{i=MD+1}^{2 \cdot MD-1} (1 \cdot prob(x(i)|\Theta_B)) \tag{4}$$

Equation 3 shows the computed likelihood given $2MD$ acoustic frames and remaining in cluster A during all of them. Equation 4, on the other hand, shows

the total likelihood if we jump to a model B after the initial MD frames. When both models are the same (A=B) it is desired that eq. 3 be greater than eq. 4 or else the possible speaker turn durations would be strongly quantized to the MD duration. In this case, it happens when $\alpha^{MD} > \frac{\beta}{M}$

Setting the values of $\alpha = 0.9$ and $\beta = 1 - \alpha$ caused long speaker turns to be artificially penalized against turns with the minimum MD frames. In order to remove this effect (since we do not have a priori information on the average turn length of the input data), we propose to set the value of $\alpha = 1.0$ and $\beta = 1.0$. Thus, once a segment exceeds the minimum duration, the HMM state transitions no longer influences the turn length; turn length is solely governed by acoustics. This creates a non-standard (but valid) HMM topology as $\alpha + \beta$ no longer sums to 1.

6 Experiments and Results

Speaker diarization experiments were conducted using the data distributed for the NIST Rich Transcription 2004 and 2005 Spring Meeting Recognition Evaluation, RT04s and RT05s ([7]). This consists of excerpts from multi-party meetings in English collected at six different sites. From each meeting, only an excerpt of 10 to 12 minutes is evaluated. Although a number of distant microphones is available for each meeting, only the most centrally located microphone (as defined my NIST as the SDM channel) was used to test the algorithms presented here. We merged the RT04s development and evaluation data to create a development set (a total of 16 meeting excerpts), used to adjust some of the parameters in the system. The RT05s evaluation data was used to validate the chosen parameters.

The metric used to evaluate the performance of the system is the same as is used in the NIST RT evaluations and is called Diarization Error Rate (DER). It is computed by first finding an optimal one-to-one mapping of reference speaker ID to system output ID and then obtaining the error as the percentage of time that the system assigns the wrong speaker label. The results given below are the time weighted DER average for the development and evaluation sets.

Although hand-made reference files were provided for each of the sets, they are at times inconsistent and therefore not very suitable to test any new algorithm. In fact, it is planned that for the RT06s evaluation systems will be scored using forced aligned reference files rather than hand-made ones. The automatically generated references are obtained by using a speech recognition system that aligns the words uttered by each speaker to the waveform, and therefore outputs the times where each speaker spoke, suitable for speaker diarization. In the present paper, we used a forced alignment generated using ICSI-SRI ASR system (see [8]). The meeting named NIST_20050412_1303 contains a telephone channel whose transcript was not provided; therefore it was not able to be fully aligned and was taken out of the test set, leaving us with 9 meetings.

In order to select the optimum values for the CCR parameters and the number of Gaussian mixtures per cluster, we did a greedy search on the parameter

space using the baseline system including the presented variation to the acoustic models. As we did not perform an exhaustive search, the resulting parameters might not be the optimal ones. The chosen values are CCR = 8 seconds/Gaussian and 5 Gaussian mixtures per initial cluster. The use of both parameters to determine the number of initial clusters sets all recordings to a range of clusters from 10 to 16, which in the meetings environment we have seen to work the best in previous publications.

Using the selected parameters, in table 1 we show the Diarization Error Rates of all presented systems, individually and in conjunction with each other, and the baseline system, both for the development data set and the test set.

System	Development set	Test set
Baseline system	18.38%	14.43 %
Speaker turn with no time restrictions	17.75%	14.49%
Complexity selection	17.23%	**11.68%**
Initial # models selection	17.59%	14.00%
Complexity + # initial models selection	**16.95%**	12.48%

Table 1. *DER for the development and test sets comparing the different proposed systems*

The baseline system is based on the system presented in [6], with model probabilities $\alpha = 0.9$ and $\beta = 0.1$. The second system introduces the change in the acoustic models to avoid the speaker turn length restrictions. All systems after the second one include such modification. The third and fourth systems correspond to each of the model parameter estimation techniques on their own, and the last system contains all of the proposed techniques.

By avoiding the speaker turn length restriction we obtain an improvement on the development set but not in the evaluation set, though it achieves almost the same result. All other systems improve the baseline results to different degrees. The best system on the development set is the one combining the three presented techniques, although the improvement over the baseline is 8% relative, smaller than the improvement obtained by the complexity selection algorithm on the test set, which is a 19% relative. This indicates the viability of the algorithms to be used on unseen recordings as all parameters had been trained using the development set.

By looking at the overall results, we can generalize that although we have the same number of parameters in the system (before we needed to define the number of Gaussian mixtures per cluster and the number of initial clusters, and now the number of mixtures and the CCR), it tunes better to the individual data sources and is more robust to changes in the show length and the structure of the acoustic data (e.g. how long and how often each speaker speaks).

7 Conclusions

In this paper, we presented three techniques for improving the acoustic modelling of a speaker diarization system based on agglomerative clustering. These techniques define the quantity of initial clusters to use, their complexity (number of Gaussian mixtures) and the topology of the models regarding duration constraints. We introduced a new parameter called the Cluster Complexity Ratio (CCR), which was used to define both the number of initial clusters and the cluster complexity, and allows models to adapt to each individual recording according to the amount of data available for clustering and the structure of the content in the recording. We showed an improvement of up to 8% on the development set and 18% on the test set, which ensures the viability of this method to be used on unseen data.

8 Acknowledgments

We would like to acknowledge the Speaker Diarization group at ICSI for their thoughtful comments and Joe Frankel, Adam Janin and Jose Pardo for their help. This work was done during Xavier Anguera's stay at ICSI within the Spanish visitors program.

References

1. D. Reynolds and P. Torres-Carrasquillo, "Approaches and applications of audio diarization," in *ICASSP'05*, Philadelphia, PA, March 2005, pp. 953–956.
2. S. Shaobing Chen and P. Gopalakrishnan, "Speaker, environment and channel change detection and clustering via the bayesian information criterion," in *Proceedings DARPA Broadcast News Transcription and Understanding Workshop*, Virginia, USA, Feb. 1998.
3. C. Wooters, J. Fung, B. Peskin, and X. Anguera, "Towards robust speaker segmentation: The ICSI-SRI fall 2004 diarization system," in *Fall 2004 Rich Transcription Workshop (RT04)*, Palisades, NY, November 2004.
4. J. Ajmera and C. Wooters, "A robust speaker clustering algorithm," in *ASRU'03*, US Virgin Islands, USA, Dec. 2003.
5. J. Ajmera, I. McCowan, and H. Bourlard, "Robust speaker change detection," *IEEE Signal Processing Letters*, vol. 11, no. 8, pp. 649–651, 2004.
6. X. Anguera, C. Wooters, B. Peskin, and M. Aguilo, "Robust speaker segmentation for meetings: The ICSI-SRI spring 2005 diarization system," in *Rich Transcription 2005 Spring Meeting Recognition Evaluation*, Edinburgh, Great Brittain, July 2005.
7. NIST rich transcription evaluations, website: http://www.nist.gov/speech/tests/rt.
8. A. Stolcke, X. Anguera, K. Boakye, O. Cetin, F. Grezl, A. Janin, A. Mandal, B. Peskin, C. Wooters, and J. Zheng, "Further progress in meeting recognition: The icsi-sri spring 2005 speech-to-text evaluation system," in *Rich Transcription 2005 Spring Meeting Recognition Evaluation*, Edinburgh, Great Brittain, July 2005.