Speech Overlap Detection in a Two-Pass Speaker Diarization System

Marijn Huijbregts[1], *David van Leeuwen*[2,3] *and Franciska de Jong*[1]

[1]University of Twente, Department of Electrical Engineering, Mathematics and Computer Science
[2]TNO Human Factors, [3]Radboud University, department of Language and Speech
{huijbreg,fdejong}@ewi.utwente.nl, david.vanleeuwen@tno.nl

Abstract

In this paper we present the two-pass speaker diarization system that we developed for the NIST RT09s evaluation. In the first pass of our system a model for speech overlap detection is generated automatically. This model is used in two ways to reduce the diarization errors due to overlapping speech. First, it is used in a second diarization pass to remove overlapping speech from the data while training the speaker models. Second, it is used to find speech overlap for the final segmentation so that overlapping speech segments can be generated. The experiments show that our overlap detection method improves the performance of all three of our system configurations.

Index Terms: Speaker diarization, speech overlap detection, Benchmark

1. Introduction

The goal of speaker diarization is to automatically segment an audio recording into speaker homogeneous regions. When the identity of each speaker is not known and even the number of speakers is unknown, it is the task of a diarization system to anonymously label each speaker in the recording and answer the question: 'Who spoke when?' [1].

Since 2004 NIST has organized evaluations of speaker diarization technology on the meeting domain [2]. At each benchmark, diarization systems are evaluated, for a number of audio recording conditions. The primary evaluation condition allows the use of audio recorded from multiple distant microphones. As an optional task, NIST also evaluates the performance of diarization systems for the condition in which the audio input comes from just a single (distant) microphone.

One of the many challenges when performing diarization on meeting recordings is how to detect and process overlapping speech. The occurrence of overlapping speech, periods of time in which multiple people are talking simultaneously, is very characteristic of meetings. For most diarization systems, the occurrence of overlapping speech makes diarization considerably more difficult.

Overlapping speech hurts the performance of speaker diarization systems such as ours, in two ways. First, because the speaker segmentation is generated by a Viterbi search, all speech will a priori be assigned to one single speaker. This means that at least half of the overlapping speech segments will be assigned incorrectly (to nobody). Second, overlapping speech acts as noise during the decision making process. Our system, the AMI system developed for RT09s, automatically generates speaker models from the audio it is processing and overlapping speech deteriorates the precision of these models.

In this paper we present the measures we took to reduce the effect of overlapping speech in the AMI speaker diarization system developed for RT09s. First, in the following section we will provide a short description of our baseline system. In section 3 we will discuss our approach in detecting and handling overlapping speech. In section 4 we will provide the experiments we performed on our development set and in section 5 we will conclude with a short discussion.

2. System description

Our speaker diarization system is based on a system originally described in [3]. The system consists of three main components: feature extraction, speech activity detection and speaker diarization. An extensive description of these components can be found in [4, 5]. In this section we will provide a short description of these components and then we will briefly describe our recent addition of a delay feature stream.

2.1. Feature Extraction

The meetings under evaluation are recorded with multiple distant microphones. The audio signal of each microphone is first passed through a Wiener filter for noise reduction. The implementation of the Wiener filtering that was used, was taken from the noise reduction algorithm developed for the Aurora 2 front-end proposed by ICSI, OGI and Qualcomm [6]. After Wiener filtering, the channels are combined into one 'enhanced' channel using delay and sum beamforming software (BeamformIt 2.0[1]). This software determines the delay of each signal relative to the other signals and removes this delay before summing all signals together [7]. From the resulting 16kHz audio file, the first nineteen Mel Frequency Cepstral Coefficients (MFCC) are extracted.

2.2. Speech activity detection

For RT07s we developed, in collaboration with ICSI, a robust speech activity detection (SAD) component that is described in [4]. This components finds all speech regions in two steps: first, using a bootstrapping speech/non-speech detection an initial segmentation is created and models for speech, silence and audible non-speech are generated. In the second step these models are applied to generate the final speech/non-speech segmentation.

2.3. Speaker diarization

The diarization component that uses the speech segments provided by the SAD component as input, is based on the use of Hidden Markov Models (HMM) with Gaussian Mixture Models (GMM) as probability density functions and is described in-depth in [5]. In this system, each speaker is represented by a

[1] www.icsi.berkeley.edu/~xanguera/beamformit

string of states that share a single GMM. Initially a high number of strings is placed in parallel in the HMM and by using agglomerative clustering, the number of strings is reduced until the correct number of speakers is reached. The final speaker segmentation is obtained by performing a Viterbi search on all audio that contains speech. All audio that is processed by the same string of states during this alignment is grouped together as speech from one speaker. By using a string of states to represent each speaker (instead of a single state), a minimum duration of each speech segment is guaranteed. The merging and stopping criteria that are needed for agglomerative clustering are based on the Bayesian Information Criterion (BIC) [8].

2.4. Delay feature stream

A by-product of the beam forming toolkit are the actual delays between microphones with which a sound is recorded. In [9] it is shown that by applying the delays as a second feature stream, the diarization error rate (DER) decreases by 21% relative. The PDF of each state is modelled by two GMMs: one for the MFCC stream and one for the delay stream. In the most recent version of this diarization system [10], the likelihoods of these two GMMs are scaled by factors, the stream weights, that are determined on basis of the reversed entropy rate of the ΔBIC scores. Using this method, the variances of the BIC scores of both streams are normalized. The initial delay feature GMM is modeled with only one gaussian. We have adopted this method with two adjustments.

First, in [10] the weights are adjusted after each merging iteration during the entire clustering process. We found that in the final iterations, when there are only a few BIC scores to compare, the weight starts shifting to the delay stream and the system tends to over-cluster. Therefore, we only determine the weights during the first two merging iterations and leave them fixed for the remaining iterations.

Second, we noticed that on average, the BIC values of the delay stream have a positive offset relative to the MFCC stream BIC values. In some cases all values are positive, making the system over-cluster. Therefore, in a similar fashion as the variance normalization of the BIC scores (the weights) we normalized the BIC scores of the delay stream to have the same mean as the MFCC stream scores in the first two clustering iterations.

3. Overlapping speech

In an earlier study ([11]) we discovered that overlapping speech is responsible for a considerable part of the diarization error rate. Overlapping speech is a problem in two ways. First, because our system is not able to output overlapping speech segments. The Viterbi segmentation only outputs the one most likely speaker at each time. Second, during the training process of the speaker models, the overlapping speech segments act as noise. Because the overlapping speech is not at all similar to the speech of the individual speakers, it only degrades the models and with that, the final segmentation.

Although overlapping speech presumably is a problem for many speaker diarization systems, the only known successful study on overlapping speech detection for speaker diarization is [12]. In this research, an overlapping speech detection system is presented that is used to assign the overlapping speech to multiple speakers in a post-processing step. Our approach will be similar, except that we will not use any training data for creating the overlap detector and we will not only use the overlapping speech knowledge in a post-processing step, but in the entire diarization process.

3.1. The approach

We assume that overlapping speech can be modeled using a single GMM, regardless of which speakers are involved. If we have such a GMM, we can add it to the HMM and perform a Viterbi run. The Viterbi search is then expected to assign the overlapping speech to the overlap model instead of to the individual speaker models. With such an overlap model we can improve the system in two ways. First, we can apply the overlap model during the final Viterbi alignment and assign all overlap segments to one or more speakers using heuristics. Second, if we apply the overlap model during the entire procedure for creating the speaker models, at each Viterbi iteration the overlapping speech will be assigned to the overlap model and the speaker models are not contaminated with overlapping speech, resulting in purer models.

It is possible that by modeling overlapping speech in the final Viterbi run, we introduce errors. The overlap model and the algorithm for assigning overlapping speech to speakers will probably not be perfect and it might be the case that the gain of modeling overlapping speech is smaller than the errors introduced by the overlap model and the algorithm.

For applying the overlap model to train purer speaker models, the overlap model does not have to be perfect. Every bit of overlap that it will take away from the models will improve them a bit. There is a risk though, of taking away too much clean data as well so that the speaker models are trained on less data and might not be trained as well as normally.

3.2. A two-pass system

Even with very good overlap models, it is hard to predict beforehand if it will be beneficial for a specific recording to apply it. We solve this problem by running the system both with and without overlap detection. For both system passes we run a final Viterbi without the use of the overlap model. Because the HMM of both systems consist of exactly the same number of Gaussians and exactly as many training iterations were used to create both HMMs, it is possible to compare the total likelihoods of the final Viterbi run of the two system passes directly. We decide whether or not to use the overlap detection on basis of these final likelihoods.

3.3. Generating overlap models

It is possible to use a training set of overlapping speech to create the overlap model, but this would mean that a new overlap model needs to be created for each new domain that the system is used on. The philosophy of our system is that because it doesn't need any training data it can be applied in new situations without the need of re-training or re-tuning, and therefore we decided to attempt to create an overlap model automatically out of each recording in the evaluation set.

In order the create this model we make a rather bold assumption. We assume that at each speaker change there is a higher probability of overlap in speech than at other moments. Using the final segmentation of the first pass (diarization without overlap detection), We train a model with five Gaussians on every final 500ms of a speech segment before a speaker change and the first 500ms after a speaker change. We then add this overlap model to the HMM and perform three Viterbi iterations, re-training the overlap model after each iteration. After each Viterbi iteration, more overlapping speech should be assigned to the overlap model (not only the overlap speech at each speaker change) that is used to train a better overlap model.

3.4. Applying overlap models

In the second diarization pass, the overlap model is applied in two ways. First, it is added to the HMM during each decoding iteration. In contrast to the speaker models, the overlap model is not used during the re-training phase, to avoid the risk that the overlap model will slowly be trained towards one of the speakers.

Second, the overlap model is used in the final Viterbi run to detect all overlapping speech segments. An algorithm is needed in order to assign these segments to two or more of the speakers. We decided on a very straightforward algorithm. The overlapping speech segments are assigned to both speakers before and after the overlapping speech segment. If the speech before and after the overlapping speech segment is from one single speaker, the overlapping segment will be assigned only to this one speaker.

With this method of speech overlap assignment we assume that overlapping speech is always produced by only two people. Although in reality this assumption is not always the case, it is true for the majority of overlapping speech (in circa 80% of the cases, [12]).

Our method of speech overlap assignment further assumes that most overlap will be due to one speaker interrupting the other. Of course this is not always the case, but because the overlap model is initially trained on speaker boundaries and is therefore biased towards interruptions, our straightforward assignment method seems a logical first approach.

4. Experiments

In this section we will discuss the experiments that we have performed on the two-pass diarization system described in the previous sections. We have performed our experiments on a test set of 27 conference meetings from earlier rich transcription evaluations. (see table 1). Before we describe our experiments on the two-pass overlapping speech system, we will discuss the performance of our baseline system on the development set.

Meeting ID	
AMI20041210-1052	AMI20050204-1206
CMU20050228-1615	CMU20050301-1415
CMU20050912-0900	CMU20050914-0900
CMU20061115-1030	CMU20061115-1530
EDI20050216-1051	EDI20050218-0900
EDI20061113-1500	EDI20061114-1500
ICSI20000807-1000	ICSI20010208-1430
NIST20030623-1409	NIST20030925-1517
NIST20051024-0930	NIST20051102-1323
NIST20051104-1515	NIST20060216-1347
TNO20041103-1130	VT20050304-1300
VT20050318-1430	VT20050408-1500
VT20050425-1000	VT20050623-1400
VT20051027-1400	

Table 1: *The 27 conference meetings that we used as test set*

4.1. The baseline system

In section 2 we described our speaker diarization system that we have developed for RT09s. The performance of this system on the development set is listed in table 2. The first row of this table shows the performance on the Single Distant Microphone (SDM) condition. In the second row the Multiple Distant Microphone (MDM) system is listed that does not make use of the delay feature stream (similar to our system at RT06s) and in the final two rows we have listed our MDM system using the delay features stream with fixed weights and with the automatically determining the weights. As can be seen from the table, adding the delay feature stream with a fixed weight improves the DER with 2.19% absolute. The automatically determined weights set-up improves the system further with 0.93% absolute to 12.9%.

Experiment	%DER
SDM	19.07
MDM, baseline	16.02
MDM, Fixed weight (at 0.98)	13.83
MDM, Automatically determined weights	12.90

Table 2: *The results of our baseline systems on the development set. The baseline MDM system does not apply the delay feature stream. The other MDM systems apply the stream with fixed stream weight (third row) or with automatically determined weights (bottom row).*

4.2. Overlapping speech

We have conducted a number of experiments to test our approach for overlapping speech detection. In this section we will discuss these experiments for three system set-ups. First we will discuss overlap detection in our SDM system. Next, we will discuss the same experiments for our MDM system that does not make use of the delay feature stream and finally we will look at the experiments for the MDM system with delay features.

For each of the three systems we have conducted five experiments. The results of these experiments are listed in table 3. For each system set-up, first the baseline result, the DER without overlap detection, is listed. After the first (baseline) diarization pass, as described in section 3, for each meeting the overlap model is generated, applied and the overlapping speech segments are assigned to the two bordering speakers. The second row in each table (overlap pass 1) lists the result after this first pass overlap detection. The third row in each table lists the result after the second diarization run where the overlap model is applied in order to generate purer speaker models. The fourth column lists the results after pass selection (see section 3) where the overlap model is only used to create pure models, but not in the final iteration to assign overlap segments to multiple speakers. In the final row, the best pass is picked and overlap detection is used for both creating better speaker models and assigning overlap segments to multiple speakers.

As can be seen from the table, overlap detection improves all three system set-ups, as long as the pass selection is used. For some meetings our overlap detection method deteriorates the result (and therefore the DER of overlap pass 1 and 2 may be higher than the baseline DER), but it is possible to select those recordings where overlap detection is useful fully automatically. The gain of overlap detection is limited for our full MDM system. In only 6 of the 27 meetings, the second pass was chosen during pass selection. From these six meetings, five actually improved the result. The DER of the sixth meeting went from 8.2% to 9.7%.

For all system set-ups, applying our straightforward method

Experiment	%Miss	%FA	%Spkr	%Total
SDM				
Baseline	5.3	2.3	11.5	19.07
Overlap pass 1	5.1	2.9	10.9	18.95
Overlap pass 2	5.1	3.0	11.8	19.92
Best pass, no overlap	5.3	2.3	10.7	18.33
Best pass	5.2	2.4	10.6	18.27
MDM, without using the delay feature stream				
Baseline	4.6	1.9	9.5	16.02
Overlap pass 1	4.4	2.5	8.9	15.78
Overlap pass 2	4.4	2.6	8.2	15.12
Best pass, no overlap	4.6	1.9	9.3	15.89
Best pass	4.4	2.4	8.5	15.29
MDM, using the delay feature stream				
Baseline	4.6	2.0	6.3	12.90
Overlap pass 1	4.5	2.1	6.3	12.90
Overlap pass 2	4.5	2.1	6.6	13.21
Best pass, no overlap	4.6	2.0	6.3	12.85
Best pass	4.6	2.0	6.2	12.83

Table 3: *The results of our overlapping speech experiments on our three systems: SDM, MDM without the use of the delay feature stream and MDM with use of delay features.*

to assign overlap regions to multiple speakers is successful. The fourth row in each table lists the result without this method. In all cases the DER of this experiment is lower than the baseline, but not as low as the final experiment where overlap detection is also applied for assignment of overlapping speech segments.

5. Discussion

In this paper we have presented our speaker diarization developed for the NIST RT09s evaluation and the measures we took to handle overlapping speech. In line with the philosophy of not using any training data for training our diarization models, we have developed a method for speech overlap detection that does not require any models trained on a training set.

As was shown in the experiments section, the addition of a second feature stream using delay features generated during delay sum beamforming, improved our system with 3.12% DER absolute on our development set. The overlapping speech detection improved all three of our system set-ups with 0.8%, 0.73% and 0.07% absolute DER respectively (a relative improvement of: 4.20%, 4.55% and 0.54%).

Compared to the two other system configurations, the DER improvement of the full MDM system is limited. The overlap pass was only selected for 6 of the 27 meetings. Judging from the other experiments, for MFCC features it is possible to create a single model for all overlapping speech. We surmise that this might not be the case for delay features where the features will have significantly different values when the overlapping speech comes from varying directions. In future work we will investigate methods to solve this problem. One solution could be to use more than one gaussian for the modeling of the delay features for overlap detection.

For all system set-ups, applying our straightforward method to assign overlap regions to multiple speakers is successful, but it is interesting to notice that although the percentage of missed speech drops when applying our method, the percentage of false alarms increases. This implies that in some cases we have detected speech overlap while it is not actually there. In future work we will investigate if it is possible to use more complex models for overlap detection in this final step so that we can reduce the false alarms. We will also examine other methods of assigning the overlapping speech to speakers, such as relying more on the delay features in these cases because during overlapping speech these features might provide more information than MFCC features.

6. Acknowledgments

This paper is partly based on research carried out in the European Union 6th FWP IST Integrated Project AMIDA (Augmented Multi-party Interaction with Distant Access, FP6-506811) and the project CHoral - Access to Oral History, which is funded by the NWO program CATCH (`http://www.nwo.nl/catch`).

7. References

[1] D. Reynolds and P. Torres-Carrasquillo, "Approaches and applications of audio diarization," Philadelphia, PA, March 2005, pp. 953–956.

[2] J. G. Fiscus, J. Ajot, and J. S. Garofolo, "The rich transcription 2007 meeting recognition evaluation," in *Multimodal Technologies for Perception of Humans*, ser. Lecture Notes in Computer Science. Berlin: Springer Verlag, 2008.

[3] J. Ajmera, H. Bourlard, I. Lapidot, and I. McCowan, "Unknown-multiple speaker clustering using HMM," in *proceedings of the International Conference on Spoken Language Processing (ICSLP)*, Denver, Colorado, USA, 2002.

[4] M. Huijbregts, C. Wooters, and R. Ordelman, "Filtering the unknown: Speech activity detection in heterogeneous video collections," in *proceedings of Interspeech*, Antwerp, Belgium, August 2007.

[5] D. van Leeuwen and M. Huijbregts, "The AMI speaker diarization system for NIST RT06s meeting data," in *Machine Learning for Multimodal Interaction (MLMI)*, ser. Lecture Notes in Computer Science, vol. 4299. Berlin: Springer Verlag, October 2007, pp. 371–384.

[6] A. Adami, L. Burget, S. Dupont, H. Garudadri, F. Grezl, H. Hermansky, P. Jain, S. Kajarekar, N. Morgan, and S. Sivadas, "Qualcomm-icsi-ogi features for asr," in *proceedings of ICSLP*, 2002.

[7] X. Anguera, "Robust speaker diarization for meetings," Ph.D. dissertation, Universitat Politecnica De Catalunya, 2006.

[8] G. Schwartz, "Estimating the dimension of a model," *The Annals of Statistics*, vol. 6, no. 2, pp. 461–464, 1978.

[9] J. M. Pardol, X. Anguera, and C. Wooters, "Speaker diarization for multiple distant microphone meetings: Mixing acoustic features and inter-channel time differences," in *proceedings of Interspeech*, 2006.

[10] C. Wooters and M. Huijbregts, "The ICSI RT07s speaker diarization system," in *Multimodal Technologies for Perception of Humans*, ser. Lecture Notes in Computer Science. Berlin: Springer Verlag, 2008.

[11] M. Huijbregts and C. Wooters, "The blame game: Performance analysis of speaker diarization system components," in *proceedings of Interspeech*, Antwerp, Belgium, August 2007.

[12] K. Boakye, B. Trueba-Hornero, O. Vinyals, and G. Friedland, "Overlapped speech detection for improved speaker diarization in multiparty meetings," in *proceedings of International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, Las Vegas, Nevada, 2008.