The IBM RT07 Evaluation Systems for Speaker Diarization on Lecture Meetings

Jing Huang, Etienne Marcheret, Karthik Visweswariah, Gerasimos Potamianos

IBM Thomas J. Watson Research Center
Yorktown Heights, NY 10598, U.S.A.

{jghg,etiennem,kv1,gpotam}@us.ibm.com

Abstract. We present the IBM systems for the Rich Transcription 2007 (RT07) speaker diarization evaluation task on lecture meeting data. We first overview our baseline system that was developed last year, as part of our speech-to-text system for the RT06s evaluation. We then present a number of simple schemes considered this year in our effort to improve speaker diarization performance, namely: (i) A better speech activity detection (SAD) system, a necessary pre-processing step to speaker diarization; (ii) Use of word information from a speaker-independent speech recognizer; (iii) Modifications to speaker cluster merging criteria and the underlying segment model; and (iv) Use of speaker models based on Gaussian mixture models, and their iterative refinement by frame-level re-labeling and smoothing of decision likelihoods. We report development experiments on the RT06s evaluation test set that demonstrate that these methods are effective, resulting in dramatic performance improvements over our baseline diarization system. For example, changes in the cluster segment models and cluster merging methodology result in a 24.2% relative reduction in speaker error rate, whereas use of the iterative model refinement process and word-level alignment produce a 36.0% and 9.2% speaker error relative reduction, respectively. The importance of the SAD subsystem is also shown, with SAD error reduction from 12.3% to 4.3% translating to a 20.3% relative reduction in speaker error rate. Unfortunately however, the developed diarization system heavily depends on appropriately tuning thresholds in the speaker cluster merging process. Possibly as a result of over-tuning such thresholds, performance on the RT07 evaluation test set degrades significantly compared to the one observed on development data. Nevertheless, our experiments show that the introduced techniques of cluster merging, speaker model refinement and alignment remain valuable in the RT07 evaluation.

1 Introduction

There are three tasks evaluated this year in the Rich Transcription 2007 (RT07) evaluation campaign [1], conducted by the National Institute of Standards and Technology (NIST): Speaker diarization (SPKR), speech-to-text (STT), and speaker-attributed STT (SASTT), a task newly introduced this year. The three are very much interconnected, with SPKR being an important pre-processing

step to STT, but also a required part of the final SASTT output. The latter is due to SASTT aiming not only to correctly transcribe spoken words, but also to identify the generically labeled speaker of these words. A better SPKR system would therefore produce better SASTT results. An additional pre-processing step in this "cascade" of tasks is speech activity detection (SAD). This has been a separate evaluation task in the RT Spring 2006 (RT06s) evaluation campaign [2], but is now considered a "mature" task and, as such, has been sunset in RT07. Nevertheless, SAD remains an important step prior to speaker diarization.

The goal of speaker diarization is to label each speech segment (as provided by the SAD pre-processing step) with speaker information. The SPKR task is therefore sometimes referred to as the "who spoke when" problem [2]. Typically, the number of speakers present is not known a-priori. Such information needs to be determined automatically in the diarization task. In recent years, significant research effort has been devoted to the problem [3–6], with progress rigorously benchmarked in NIST speech technology evaluations [7].

There exist two main approaches to speaker diarization: The first is a bottom-up approach, i.e. hierarchical, agglomerative clustering [8–10], and the second is top-down, employing evolutive hidden Markov models (E-HMMs) [4], starting with one speaker and detecting and adding speakers in succession. Agglomerative clustering generally involves several steps: Initially, speech segments, as determined from SAD output, are investigated for possible speaker change points [11]. The output of change point detection is then fed into a speaker clustering procedure. Clustering stops when a predetermined criterion is satisfied (for example a drop in overall data likelihood from a merge). The limitation of this approach is that errors in the first two steps carry over to the final clustering step.

An improvement is to jointly optimize segmentation and clustering using an iterative procedure based on Gaussian mixture models (GMMs) of each cluster [12]. Recently, ICSI proposed purification algorithms for the iterative segmentation scheme to improve performance. Impure segments are removed before the cluster merging step, and impure frames are removed from GMM training and cluster merging [8]. LIMSI proposed to use speaker identification combined with the Bayesian information criterion (BIC) to improve performance [9, 13]. However, this approach may not work well in lecture data, where many of the speakers (audience members asking questions) do not have enough data to generate reliable speaker models. The same problem occurs with the E-HMM scheme, where speaker models are needed. This approach usually detects the most dominant speakers well, but misses speakers with little data [4].

Although it is allowed to use STT system output to assist speaker diarization in the NIST RT evaluation, most submitted systems do not take advantage of word output from the STT task [2]. To our knowledge, such information has in the past been exploited by LIMSI, for example use of spoken cues ("Back to you, Bob"), as a means to add information to the diarization output for Broadcast News data [14], as well as for removal of short-duration silence segments when training speaker models [9].

In this paper, we present details of our SPKR system evaluated in the RT07 campaign for the lecture meeting domain. This represents the first year that the IBM team participated in the RT SPKR evaluation, although a baseline system has been developed last year as an STT pre-processing step [16] – but not officially evaluated. However, a separate SAD system had been evaluated [15]. A number of modifications have been made to these systems, resulting in the RT07 IBM SPKR system. In summary:

- A simpler SAD algorithm is employed, compared to the one in RT06s [15]. It is based on a speech/non-speech HMM decoder, set to an optimal operating point for missed speech / false alarm speech on development data. Because missed speech cannot be recovered in following speaker segmentation steps, the operating point is selected to miss only a small amount of speech, but at the same time not to introduce too many false alarms.
- Word information generated from STT decoding by means of a speaker-independent acoustic model is used to improve speaker clustering. Such information is useful for two reasons: It filters out non-speech segments, and it provides more accurate speech segments to the speaker clustering step, removing short silence, background noise, and vocal noise that do not discriminate speakers and cause overlaps of cluster models. As a result, only speech frames are used to train and compare cluster models.
- GMM-based speaker models are built from an available segmentation (for example, as provided by SAD), and the labels of each frame are refined using these GMM models, followed by smoothing the labeling decision with its neighbors. A result of re-classification and smoothing is the possibility that the original segments can be further segmented, in effect locating speaker change points within the initial segmentation. This process is significantly better than last year's change point detection approach.

The rest of the paper is organized as follows: Section 2 briefly overviews the baseline SAD and speaker diarization systems developed in RT06s. Section 3 presents RT07 modifications to the components of the baseline systems to improve diarization performance. Section 4 describes two system level variations taking advantage of the improved components. Section 5 is devoted to the experimental study and discussions, and Section 6 concludes the paper.

2 Baseline SAD and SPKR Systems

Speech activity detection (SAD) is a prerequisite to both SPKR and STT. After SAD, long segments of non-speech (silence or noise) are removed, and the audio is partitioned into shorter segments for fast decoding and speaker segmentation. For the RT06s evaluation, the IBM team developed two SAD systems: The one was officially evaluated, and it was based on a complex scheme of fusing acoustic likelihood and energy features for modeling three classes by full-covariance GMMs. During testing, the classes were collapsed into speech and silence, and appropriately smoothed to yield the final SAD result. Significant performance

gains were observed when combining SAD results across multiple far-field channels by simple "voting" (decision fusion) [15].

The second scheme was employed as a first step in the IBM RT06s STT system, but was not evaluated separately [16]. We use this scheme as the first step for our SPKR/STT development this year. In more detail, it was an HMM-based speech/non-speech decoder; speech and non-speech segments were modeled with five-state, left-to-right HMMs. The HMM output distributions were tied across all states and modeled with a mixture of diagonal-covariance Gaussian densities. The non-speech model included the silence phone and three noise phones. The speech model contained all speech phones. Both were obtained by applying a likelihood-based, bottom-up clustering procedure to the speaker-independent acoustic model developed for STT, but adapted to the CHIL part of the training data by maximum a-posteriori (MAP) adaptation.

Our baseline speaker diarization system was originally developed for the EARS transcription system [17]. The framework is similar to the one described in [10], and it's briefly summarized here: All homogeneous speech segments as determined by the SAD output were modeled using a single Gaussian density function with diagonal covariance, and were bottom-up clustered into a pre-specified number of speaker clusters using K-means and a Mahalanobis distance measure. This distance measure between two D-dimensional Gaussians of diagonal covariance, denoted by $N(\mu_k, \sigma_k)$, $k = i, j$, is given by

$$dist(i,j) = \sum_{d=1}^{D} \frac{(\mu_i(d) - \mu_j(d))^2}{(\sigma_i^2(d) + \sigma_j^2(d))} \ . \tag{1}$$

For CHIL data, the number of speaker clusters was set to four for each lecture. This particular scheme proved sufficient for STT, but was never evaluated as a separate SPKR system in RT06s. For both SAD and SPKR tasks, 24-dimensional PLP acoustic features were used.

3 Improvements over Baseline Systems

We now proceed with details of the improvements introduced in RT07 to our SPKR system.

3.1 Improvement on SAD

By varying the number of Gaussians for speech and non-speech models, we are able to obtain different operating points of SAD performance, i.e. the ratio of missed speech vs. false alarm speech. For example, if too many Gaussians are used for speech, then the false alarm rate becomes high; if too many Gaussians are used for non-speech, then the miss rate grows. Because the missed speech cannot be recovered in the following speaker segmentation step, we choose an operating point that would only miss a very small amount of speech, but at the same time would not introduce too many false alarms. In fact, this simple

scheme works extremely well. Interestingly, at that operating point, no gain is obtained by combining multiple distant microphone SAD outputs, in contrast to the RT06s system [15]. The output of SAD is purified as discussed next.

3.2 Incorporating Word Output Alignments

In general, the speaker diarization task is performed before decoding. Here we propose a slight variation, namely to use the decoded output from a speaker-independent acoustic model in order to further refine SAD output, prior to its use in the speaker diarization step. The information is used in two ways:

- We remove segments with only silence, background noise, and vocal noise. These are segments that SAD failed to identify as non-speech.
- In the subsequent speaker clustering steps, we ignore frames that correspond to silence, background noise, and vocal noise.

The first constitutes a segment-based purification, whereas the second is frame-based purification [8]. By identifying non-speech frames and removing them from the speaker model training step, one expects better speaker clustering.

3.3 Clustering and Refinement

An important change from our baseline SPKR system is that instead of using a fixed number of desired speakers, we estimate the initial number of speakers according to a minimum number of expected frames. This process results in an upper bound on the expected number of speakers. Subsequently, speaker clustering and refinement reduce this number, as discussed in this section.

An additional change has to do with the employed acoustic features. Instead of 24-dimensional PLPs, we switch to 19-dimensional MFCCs, with the energy term dropped. Such features constitute the traditional feature space used for speaker recognition.

A crucial step in the whole process is the maximum-likelihood based clustering and GMM refinement. This consists of the following steps:

- *Initialization:* To build the initial speaker models we use the K-means process as follows: We take the segments sequentially partitioned equally to each speaker. A single-mixture full-covariance (FC) Gaussian for each speaker model is then estimated using a maximum likelihood criterion on each of these segments. In parallel, segments generated by the SAD output are modeled by a single mixture FC model. Subsequently, the speaker models are re-estimated by maximum likelihood using the SAD segment sufficient statistics. In more detail, each SAD segment model is assigned to the best speaker model according to a maximum per-frame log-likelihood criterion; i.e., for speaker model i and SAD segment model j , the following are used

$$\begin{aligned} LL_1(i,j) &= \Sigma_{SAD(j)} + (\mu_{SPK(i)} - \mu_{SAD(j)})(\mu_{SPK(i)} - \mu_{SAD(j)})^T \\ LL(i,j) &= -\frac{1}{2}\, trace\,(\Sigma^{-1}_{SPK(i)} LL_1(i,j)) - \frac{1}{2}\log|\Sigma_{SPK(i)}|\,, \end{aligned} \tag{2}$$

where (μ, Σ) denote Gaussian model mean and full covariance. The assignment and re-estimation steps are run for ten iterations.

- *Cluster Merging:* At each step in the bottom-up clustering process, we combine the two nodes that result in the smallest likelihood loss, if merged. We stop, when no two nodes can be combined with a loss smaller than a pre-set threshold. The likelihood loss associated with merging clusters i and j is

$$dist(i,j) = N\,(\log\,|\Sigma| \,-\, p_i \log|\Sigma_i| \,-\, p_j \log|\Sigma_j|\,)\ , \tag{3}$$

  where $N = n_i + n_j$ is the total number of frames assigned to clusters i and j, and the priors on clusters are determined as $p_i = n_i/N$. Therefore, at each step in the merging process, the smallest $dist(i,j)$ determines which clusters to merge. The result is then compared against a pre-specified threshold λ, causing the merging process to be terminated if $dist(i,j) > \lambda$.
- *Refinement:* From the merging step we have frame-level assignments to each of the remaining speaker models. Using these indices, we build diagonal-covariance GMMs with ten mixtures (an empirically determined number). This is accomplished by clustering and splitting, running expectation-maximization steps between splits. From the resulting refined models we then compute the per-frame likelihoods. The frame-level likelihoods are subsequently smoothed over a 150 msec window (± 75 msec), and the frame is assigned to the appropriate model, according to the maximum score. With these new frame-level assignments the entire refinement process can be rerun. We find that after two iterations frame-level assignments stabilize.

4 SPKR System Variations Used

In our experiments, in addition to the baseline, we consider two similar SPKR systems – referred to as "IBM 1" and "IBM 2" – that adopt most of the above improvements. In particular, both take advantage of improved SAD and word-level alignment and employ sequential cluster pre-initialization. The difference lies in the clustering metric and the use of the secondary GMM refinement step:

- *IBM 1:* During cluster merging, the Mahalanobis distance metric (1) is used on the over-segmented input, instead of (3). The merging process is terminated when a pre-specified threshold value is reached – determined on development data. Use of the word-level alignment information results in system "IBM 1 + align". Note that the Mahalanobis distance metric of equation (1) is used in both the SAD segment assignment to a speaker cluster and the distance metric, when deciding which clusters to merge.
- *IBM 2* is generated by the initialization and cluster merging steps, as described in Section 3.3. We denote use of word-level alignments and the GMM refinement step by "IBM 2 + align" and "IBM 2 + refine", respectively.

5 Experiments and Results

Our experiments are conducted on the lecture meeting data collected by five partners of the CHIL consortium ("Computers in the Human Interaction Loop")

Table 1. Overall diarization error (DER), %, of the baseline and improved (tuned) speech activity detection (SAD) systems on our development set based on input from a single distant microphone (SDM). The two systems use different numbers of Gaussians to model speech and silence. DER break-down into its two components (missed and false alarm errors) is also shown.

systems	missed (%)	false alarm (%)	DER (%)
sad.16.16 (baseline)	0.3	16.5	16.8
sad.100.32 (improved)	1.3	3.0	4.3

[18]. For development data, necessary for SAD and cluster merging threshold tuning, we use part of the RT06s evaluation test set containing 28 lecture meetings recorded in CHIL smart rooms. From this set, 27 lectures are used as the development set, with one lecture excluded due to it being closer to the so-called "coffee-break" scenario [1]. Of course, for final system evaluation the lecture meeting part of the RT07 test set is used. Note that all experiments in this section are reported for the single-distant microphone (SDM) condition, as specified in the NIST evaluation plan [1].

In accordance to NIST scoring, results are reported in terms of diarization error rate (DER). DER is calculated by first finding the optimal one-to-one mapping between reference speakers and the hypothesized ones, and then computing the percentage of time that is wrongly assigned according to the optimal mapping. DER includes speaker error time, missed speaker time, and false alarm speaker time, thus also taking SAD errors into account [2]. SAD performance is measured in DER as well, but with all speakers labeled simply as speech. Segments with overlapping speech are also included in the DER computation.

5.1 Development Set SAD and SPKR Results

As already mentioned, there exists a tradeoff in SAD system performance between false accepts and missed speech, with the operating point being a function of the number of Gaussians modeling the speech and silence classes. In our reported experiments, we imply this dependence by denoting the SAD systems as "sad . < number of speech Gaussians > . < number of silence Gaussians >". In particular, the baseline system is "sad.16.16" and the improved (after tuning) is "sad.100.32". Table 1 compares performance of these two systems at the SDM condition. Notice the significant reduction in false alarm rate with a small only impact on missed speech over the baseline. The overall SAD DER is reduced by 74.4% relative.

We now proceed to report experiments on the speaker diarization system. We first investigate the impact on DER of the cluster merging threshold. Fig. 1 illustrates results for three configurations of the "IBM 2" system, namely "IBM 2", "IBM 2 + align", and "IBM 2 + refine". Not depicted is the "IBM 2 + align + refine" system, as no gain was achieved using alignment following refinement. This figure demonstrates that the optimal cutoff threshold lies around 7000.

Table 2 provides a comparison between the baseline SPKR system (the crude RT06s system that uses a pre-set number of speaker clusters) and the "IBM

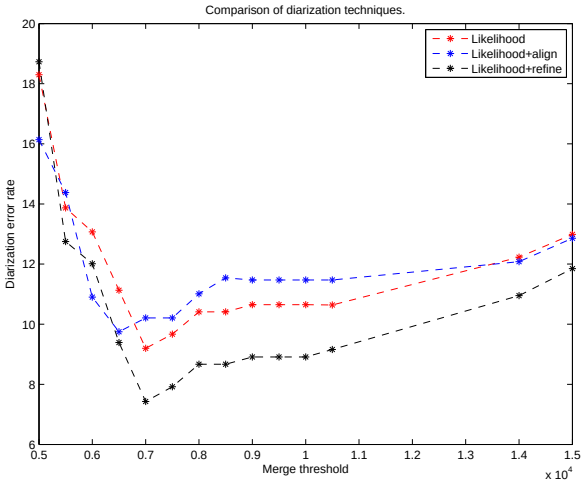

Fig. 1. Impact of the cluster merging threshold on DER on our development set, depicted for three variants of the "IBM 2" speaker diarization system.

1" and "IBM 2" systems, under various configurations concerning the use of alignment and refinement, but in all cases at their respective optimal cluster merging thresholds. Note that the cluster merging thresholds differ, due to the use of different distance metrics (see (1) and (3)). Unfortunately, due to lack of time, we did not explore the "IBM 1 + refine" system.

Finally, Fig. 2 illustrates the effect of SAD error rate (missed plus false alarm errors) on SPKR DER. In this experiment, we increase SAD error from the 4.3% value of Table 2 to 12.3%, by switching to a "sad.256.16" system. Notice that the average reduction in speaker error rate by employing the selected "sad.100.32" system is 20.3% over the use of "sad.256.16". In particular, at the optimal cluster merging threshold of 7000, we observe a 56.7% relative reduction in speaker error rate (from 7.4% to 3.2%).

5.2 Evaluation Results

The IBM team submitted the following systems relevant to speaker diarization for the RT07 evaluation:

Table 2. DER and its break-down, %, for various SPKR systems measured on development data, depicted at their optimal cluster merging thresholds (if applicable).

systems	opt. thresh.	missed (%)	false alarm (%)	speaker error (%)	DER (%)
IBM baseline	—	0.3	16.5	53.3	70.1
IBM 1	0.6	1.3	3.0	6.6	10.9
IBM 1+align	0.6	1.3	3.0	5.6	9.9
IBM 2	7000	1.3	3.0	5.0	9.3
IBM 2+align	6500	1.3	3.0	5.5	9.8
IBM 2+refine	7000	1.3	3.0	3.2	7.5

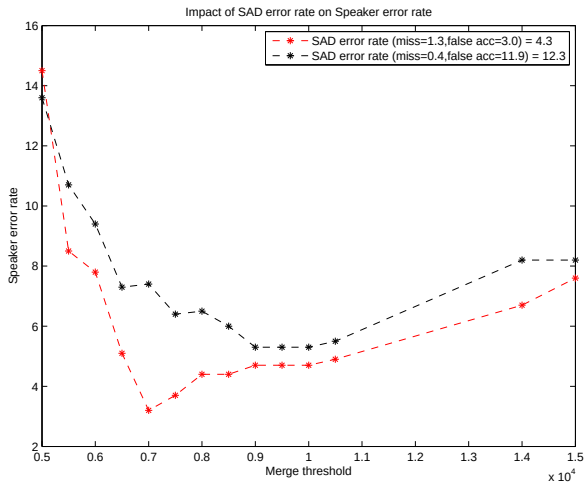

Fig. 2. Speaker error rate on development data as a function of the cluster merging threshold for two different SAD systems with error rates of 4.3% and 12.3%.

- *Speaker Diarization (SPKR) task:* With the speaker diarization deadline a week before SASTT was due, and owing to the fact that the "IBM 2" system was not ready by the first deadline, for SPKR we submitted the "IBM 1 + align" system. Threshold tuning was applied per recording site, tuned on the development set.
- *Speaker-Attributed Speech-To-Text (SASTT) task:* By the SASTT deadline, the "IBM 2 + refine" system was ready. Results from this system were used in SASTT with a cluster merging threshold of 7000.
- *Speech Activity Detection (SAD) subsystem:* The "sad.100.32" system was used in both cases.

We now proceed to report evaluation set results using the above systems. Table 3 summarizes experiments on the 32-segment RT07 test set. We observe that the overall SAD performance (missed and false alarm errors) remains relatively consistent, degrading from 4.3% on development data to 6.3% on the evaluation set. On the other hand, speaker diarization performance does not show similar stability. It is clear from the table that the chosen thresholds in our submitted systems (SASTT and SPKR), tuned on basis of development data, do not generalize well to the test set. In particular, it seems that our primary system submitted to the SPKR task ("IBM 1 + align") gets penalized by the selection of site-specific thresholds – as compared to the use of a site-independent threshold of 0.6 determined on development data (see also Table 2). In general, it also seems that this system is slightly less sensitive to the tuned threshold than the "IBM 2 + refine" system, submitted to the SASTT task. Nevertheless, the "IBM 2 + refine" system has the potential to generate a lower DER – if only the optimal threshold of 15000 were used!

It is interesting to note that the "IBM 1 + align" system improves performance over its "IBM 1" variant for both development and evaluation data,

Table 3. DER and its break-down, %, on the RT07 test set for the two IBM systems in various configurations, and for various thresholds tuned on development or evaluation data. The latter are marked with "opt.", and of course constitute a "cheating" experiment. The systems in bold are the ones officially benchmarked in RT07.

systems	threshold	missed	false alarm	speaker error	DER
IBM 1	0.6	2.4	3.9	21.9	28.2
IBM 1	0.9 (opt.)	2.4	3.9	18.5	24.8
IBM 1+align	site-spec. (RT07 SPKR)	2.4	3.9	23.7	30.0
IBM 1+align	0.6	2.4	3.9	21.0	27.3
IBM 1+align	0.85 (opt.)	2.4	3.9	18.7	25.0
IBM 2	7000	2.4	3.9	24.8	31.1
IBM 2	15000 (opt.)	2.4	3.9	17.6	23.9
IBM 2+refine	7000 (RT07 SASTT)	2.4	3.9	21.4	27.7
IBM 2+refine	15000 (opt.)	2.4	3.9	16.5	22.8

achieving a relative 9.2% DER reduction (from 10.9% down to 9.9% – see Table 2), and 3.2% (28.2% to 27.3%) respectively. In terms of pure speaker error rate, the relative gains are 15.2% (6.6% to 5.6%) and 4.1% (21.9% to 21.0%), respectively. Similar improvements occur for the "IBM 2 + refine" system over the "IBM 2" one: Namely, for DER, a 19.4% (9.3% becomes 7.5%) and a 10.9% (31.1% to 27.7%) relative reduction are observed on the development and evaluation sets, respectively; in terms of pure speaker error rates, these reductions become even more pronounced, at 36.0% (5.0% to 3.2%) and 13.7% (24.8% to 21.4%) relative. Comparing the baseline IBM 1 and IBM 2 systems, we see a 24.2% (5.6% to 4.0%) relative reduction in speaker error rate on the development test set. On the evaluation test set at optimal thresholds we see a smaller 4.9% (18.5% to 17.6%) relative reduction between the two baseline systems.

This "picture" changes though, if threshold tuning is performed on the evaluation set. Under such scenario, the "IBM 1 + align" shows a 1.1% degradation in speaker error over the "IBM 1" system. Furthermore, the relative DER gain of the "IBM 2 + refine" over the "IBM 2" system is reduced to only 4.6% (23.9% to 22.8%), corresponding to a 6.3% relative gain in speaker error (17.6% to 16.5%).

It is worth making two final remarks. The one is that the DER numbers reported in Table 3 deviate somewhat from the RT07 results reported by NIST for the IBM system. In particular, the official DER number for the RT07 SPKR system is 29.83%, contributed by 2.5% missed, 3.6% false alarm, and 23.7% speaker errors. The reason for the discrepancy in the results is unclear to us, but is most likely due to the scoring software. The second remark has to do with the multiple distant microphone (MDM) condition. In our submitted SPKR MDM system, we have not performed any microphone channel combination (for example, signal-based or decision fusion). Instead, we used a single microphone, selected based on the highest signal-to-noise ratio among the available table-top microphones in each lecture. This turns out to be in almost all cases identical to the channel selected by NIST for the SDM condition. As a result, MDM SPKR performance is very close to the SDM one, exhibiting a DER of 30.00%.

6 Conclusions and Future Work

In this paper, we presented the IBM team efforts to improve speaker diarization (SPKR) for lecture meeting data, as part of the RT07 evaluation campaign. We first described the speech activity detection (SAD) subsystem that constitutes a greatly improved version of the one used in conjunction with the IBM speech-to-text (STT) system in the RT06s evaluation. The improvements resulted in a 74.4% relative reduction in SAD error on development data, by appropriately tuning the balance between the number of Gaussians used to model the speech and silence classes. The SAD error rate generalized relatively well from development to evaluation data, achieving a 6.3% "diarization" error on the latter.

For speaker diarization, we developed a new approach that over-segments SAD output, and subsequently initializes, merges, and refines speaker clusters. This results in a varying number of final speakers, as opposed to our simple approach employed as part of the RT06s STT system that derived a pre-set number of speakers. In the newly developed system, we employed different distance metrics (Mahalanobis, likelihood-based) and Gaussian models (with diagonal or full covariance) in the initialization and cluster merging steps, giving rise to two slightly different systems that were submitted to the SPKR and SASTT tasks. The systems dramatically improved performance over our RT06s baseline, achieving diarization errors as low as 7.5% on development data. However, these results do not carry over to the evaluation set, due to system sensitivity to the cluster merging threshold. Indeed, diarization error hovers in the range of 28% and 30% for the two systems.

To reduce such instability, one possibility is to use the modified BIC score [8], which would hopefully remove the need for such threshold altogether. Replacing the refinement step with a speaker identification system, iteratively refining the enrollment against a relevant universal background model may also help. Finally, we would like to investigate the use of multiple microphone input to improve SKPR system performance over the single channel system.

Acknowledgments

This work was supported by the European Commission under integrated project CHIL, "Computers in the Human Interaction Loop," contract no. 506909.

References

1. NIST 2007 Spring Rich Transcription Evaluation. [Online] Available: `http://www.nist.gov/speech/tests/rt/rt2007/index.html`
2. J.G. Fiscus, J. Ajot, M. Michel, and J.S. Garofolo, "The Rich Transcription 2006 Spring meeting recognition evaluation," *Machine Learning for Multimodal Interaction*, S. Renals, S. Bengio, and J.G. Fiscus (Eds.), vol. LNCS 4299, pp. 309–322, Springer, 2006.

3. X. Anguera, C. Wooters, and J.M. Pardo, "Robust speaker diarization for meetings: ICSI RT06S meetings evaluation system," *Machine Learning for Multimodal Interaction*, S. Renals, S. Bengio, and J.G. Fiscus (Eds.), vol. LNCS 4299, pp. 346–358, Springer, 2006.
4. C. Fredouille and G. Senay, "Technical improvements of the E-HMM based speaker diarization system for meeting records," *Machine Learning for Multimodal Interaction*, S. Renals, S. Bengio, and J.G. Fiscus (Eds.), vol. LNCS 4299, pp. 359–370, Springer, 2006.
5. X. Zhu, C. Barras, L. Lamel, and J.-L. Gauvain, "Speaker diarization: From Broadcast News to lectures," *Machine Learning for Multimodal Interaction*, S. Renals, S. Bengio, and J.G. Fiscus (Eds.), vol. LNCS 4299, pp. 396–406, Springer, 2006.
6. D.A. van Leeuwen and M. Huijbregts, "The AMI speaker diarization system for NIST RT06s meeting data," *Machine Learning for Multimodal Interaction*, S. Renals, S. Bengio, and J.G. Fiscus (Eds.), vol. LNCS 4299, pp. 371–384, Springer, 2006.
7. NIST Rich Transcription Benchmark Tests. [Online] Available: `http://www.nist.gov/speech/tests/rt`
8. X. Anguera, C. Wooters, and J. Hernando, "Purity algorithms for speaker diarization of meetings data," in *Proc. Int. Conf. Acoustic Speech Signal Process. (ICASSP)*, vol. 1, pp. 1025–1028, Toulouse, France, 2006.
9. X. Zhu, C. Barras, S. Meignier, and J.-L. Gauvain, "Combining speaker identification and BIC for speaker diarization," in *Proc. Interspeech*, pp. 2441–2444, Lisbon, Portugal, 2005.
10. D.A. Reynolds and P. Torres-Carrasquillo, "Approaches and applications of audio diarization," in *Proc. Int. Conf. Acoustic Speech Signal Process. (ICASSP)*, vol. 5, pp. 953–956, Philadelphia, PA, 2005.
11. J. Ajmera and C. Wooters, "A robust speaker clustering algorithm", in *Proc. Automatic Speech Recogn. Understanding Works. (ASRU)*, St. Thomas, US Virgin Islands, 2003.
12. J.-L. Gauvain, L. Lamel, and G. Adda, "Partitioning and transcription of Broadcast News data," in *Proc. Int. Conf. Spoken Language Systems (ICSLP)*, Sydney, Australia, 1998.
13. R. Sinha, S.E. Tranter, M.J.F. Gales, and P.C. Woodland, "The Cambridge University March 2005 speaker diarisation system," in *Proc. Interspeech*, pp. 2437–2440, Lisbon, Portugal, 2005.
14. L. Canseco-Rodriguez, L. Lamel, and J.-L. Gauvain, "Speaker diarization from speech transcripts," in *Proc. Int. Conf. Spoken Language Systems (ICSLP)*, Jeju Island, S. Korea, 2004.
15. E. Marcheret, G. Potamianos, K. Visweswariah, and J. Huang, "The IBM RT06s evaluation system for speech activity detection in CHIL seminars," *Machine Learning for Multimodal Interaction*, S. Renals, S. Bengio, and J.G. Fiscus (Eds.), vol. LNCS 4299, pp. 323–335, Springer, 2006.
16. J. Huang, M. Westphal, S. Chen, O. Siohan, et al., "The IBM Rich Transcription Spring 2006 speech-to-text system for lecture meetings," *Machine Learning for Multimodal Interaction*, S. Renals, S. Bengio, and J.G. Fiscus (Eds.), vol. LNCS 4299, pp. 432–443, Springer, 2006.
17. S.F. Chen, B. Kingsbury, L. Mangu, D. Povey, et al.. "Advances in speech transcription at IBM under the DARPA EARS program," *IEEE Trans. Speech Audio Language Process.*, 14(5): 1596–1608, 2006.
18. CHIL: "Computers in the Human Interaction Loop", [Online] Available: `http://chil.server.de`