Digital Signal Processing **10**, 75–92 (2000)
doi:10.1006/dspr.1999.0362, available online at http://www.idealibrary.com on IDEAL®

Speaker Recognition on Single- and Multispeaker Data

Frederick Weber, Barbara Peskin, Michael Newman, Andrés Corrada-Emmanuel, and Larry Gillick

Dragon Systems, Inc., 320 Nevada Street, Newton, Massachusetts 02460

E-mail: Fred_Weber@dragonsys.com, Barbara@dragonsys.com, MikeN@dragonsys.com, Larry@dragonsys.com

Weber, Frederick, Peskin, Barbara, Newman, Michael, Corrada-Emmanuel, Andrés, and Gillick, Larry, Speaker Recognition on Single- and Multispeaker Data, *Digital Signal Processing* **10** (2000), 75–92.

We discuss Dragon Systems' approach to the NIST Speaker Recognition tasks. For the one-speaker task, we employ a combination of methods: a basic GMM system and two LVCSR-based systems, one using standard mixture models and the other using nonparametric techniques. We discuss some explorations of the recently introduced two-speaker tasks based on the GMM system alone. "Cheating" tests using NIST-supplied keys lead us to some improvements in channel normalization, and illuminate the roles that speaker segmentation and segment selection play in these tasks.

Key Words: speaker recognition; speaker tracking; speaker segmentation; GMM; LVCSR; nonparametric density estimation.

INTRODUCTION

The problem of identifying the speaker in a short excerpt from a telephone conversation may seem a simple matter for most people if the caller is familiar, but it is a challenging one for computer-based systems. Dragon Systems has long maintained that the best way to perform automatic speaker recognition—as well as other information extraction tasks such as topic identification or keyword spotting—is through the use of a full large vocabulary continuous speech recognition (LVCSR) system. We believe that LVCSR systems provide a natural framework for encoding speech characteristics and for capturing contextual information invaluable for correctly identifying regions of interest, whether keywords or speaker intervals.

Beginning in the early 1990s, Dragon has been developing a family of speaker ID systems which put this philosophy into practice (see, e.g., [1]). We currently

1051-2004/00 $35.00

employ a basic LVCSR-based system, a contrasting Gaussian mixture model (GMM) system, and a relatively recent addition which starts with the same LVCSR front-end as our original system but uses nonparametric methods to score frame sequences. Section 1 reviews each of these systems and compares their performance and their potential for system combination on the one-speaker task, using data from several years of NIST Speaker Recognition Evaluations.

In Section 2 we report preliminary results on NIST's new two-speaker detection and tracking tasks. For these initial explorations we have used the simplest of our speaker ID systems—the GMM system—since it provides the fastest turn-around for exploratory work. We present the system used for the two-speaker tasks in the NIST 1999 evaluation and describe a series of diagnostic studies that illuminate some of the issues faced in moving from single-speaker to multispeaker data.

1. DRAGON'S SPEAKER RECOGNITION SYSTEMS

To explore the speaker recognition problem, Dragon has developed a family of three speaker ID systems. The following sections describe each system in some detail and discuss their performance on data from various evaluations.

1.1. The LVCSR System

Progress on Dragon's LVCSR-based speaker ID system has been regularly reported, e.g., [2, 3]. The system uses a basic speech recognizer to transcribe the speech stream and align the speech to the transcript, providing a frame-by-frame phonetic label. Using these labels, it then scores a general background model and a speaker-adapted target model against the speech stream and bases its speaker decision on the normalized difference of the two scores. We provide some additional details here.

For the purposes of speaker recognition we use a somewhat simplified version of Dragon's standard Switchboard-based speech recognizer [4] to transcribe the training and test data. The front-end produces a channel-normalized 44-parameter feature vector every 10 ms, consisting of eight spectral parameters, 12 cepstral parameters, and the first and second cepstral differences, which is then reduced to a 24-dimensional vector via an IMELDA transformation [5].

The recognition models are decision-tree clustered triphone models with about 12,000 output distributions, trained from roughly 60 hours of Switchboard-I conversations. Each output distribution is a mixture of up to 16 Gaussian components. The recognizer uses a simple bigram language model trained from almost 3 million words of Switchboard text. This recognizer achieves a word error rate of 46.6% on a standard test of Switchboard conversations (the "CAIP" set). This is considerably worse than our best Switchboard system (which achieves a 27.2% word error rate on CAIP), but the recognizer used here does not include such standard features as rapid

speaker adaptation, vocal tract length normalization, and more sophisticated interpolated trigram language models, partly for simplicity and speed of decoding and partly because features such as rapid adaptation are not robustly estimated on the small samples of speech available in speaker ID evaluations. We estimate that the recognizer currently used for speaker ID operates with a word error rate in excess of 50% on the short concatenated excerpts which characterize NIST evaluation data and using the tight thresholds which speed decoding to three times real-time. (Unfortunately, we cannot compute word error rate directly since correct transcriptions are not available for evaluation data.)

The transcribed, time-aligned recognizer output is scored using a different set of models and different signal processing. For speaker ID scoring, we use a 38-dimensional feature vector consisting of 19 cepstra and their first differences. Higher-order cepstra capture more speaker-sensitive information which is useful for speaker recognition but damaging for speaker-independent transcription, as we have found empirically. The use of the higher cepstra for speaker ID has been discussed in detail in [6], and their effect on our own speaker ID system is covered in [3]. The scoring models are monophone, rather than triphone, so that adaptation can cover as many output distributions as possible given the limited amount of speaker training data available. The background model is trained from about 35 hours of Switchboard data (the legal training from the 1996 NIST Speaker Evaluation) and consists of 124 output distributions, corresponding to 43 phonemes with typically three states per phone. Each output distribution is a mixture of up to 128 Gaussian components for a total of about 15,000 Gaussians.

Target models are trained via Baum–Welch adaptation [7, 8] of the background model, based on the recognizer's transcription of the speaker training data. At test time, the transcribed, time-aligned test data is scored against background and adapted target models and the average score difference is computed over all nonsilence frames. These raw scores are then corrected for speaker and handset using HNORM, as in [9].

The LVCSR approach provided state-of-the-art performance in the early years of NIST evaluations (even with much higher word error rates than today), and it continues to perform well given sufficient training and test material (e.g., 2 minutes of training data and 30 seconds of test data). But as the evaluations have come to focus on shorter test segments, the performance of the LVCSR system has lagged. We therefore began examining other approaches to the speaker recognition task.

1.2. The GMM System

To explore the strengths and weaknesses of our LVCSR-based approach, Dragon built a parallel GMM-based system in 1996. The GMM system was intended to duplicate as far as possible the scoring of the LVCSR approach, but without LVCSR: it uses the same 38-dimensional feature vectors, training protocol, normalization techniques, etc.

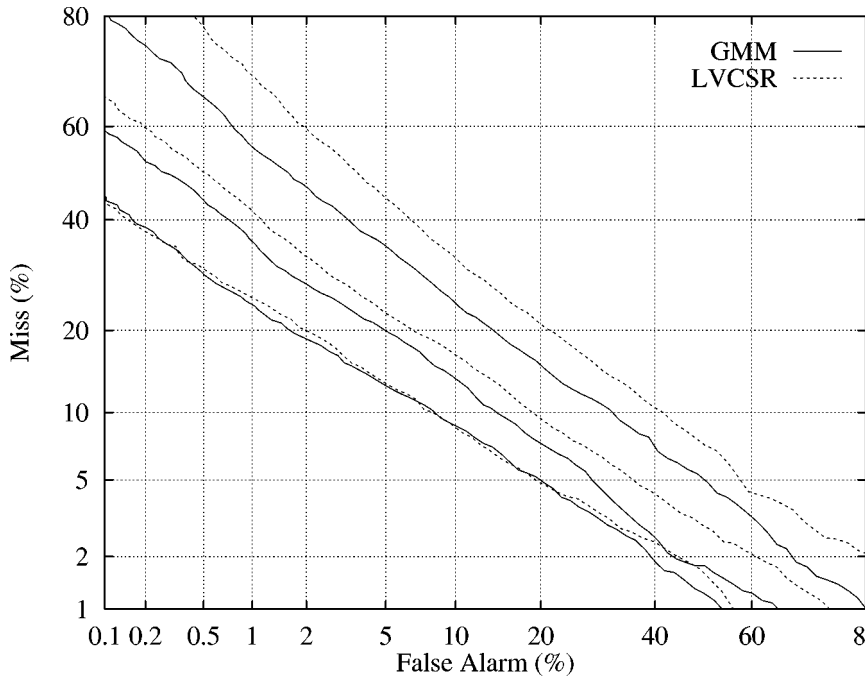

FIG. 1. Results on the male data set from the February 1998 NIST evaluation, comparing the results of our LVCSR and GMM systems on the 3- (top pair), 10- (middle), and 30-sec (bottom) samples.

The GMM system is modeled on the system described by MIT Lincoln Lab in [10]. A universal background model consisting of a single 2048-component Gaussian mixture model is trained from about an hour of speech from the 1996 development set. As in our LVCSR system, target models are trained by adapting this speaker-independent background model: we aggressively update component means, weakly adapt component deviations, and leave mixture weights unchanged. The GMM system requires no recognition stage; we simply score every frame of speech exceeding an energy threshold and compute the average score difference (normalized via HNORM) over these frames. To speed up the scoring given so many mixture components, we introduce a "decoding" pass where we identify the five best-scoring components in each frame. In subsequent passes only these top five components are rescored against adapted and unadapted models. This makes further runs on the data very fast; in particular, the incremental cost of scoring an additional speaker's model on the test data is negligible.

Comparative performance of the GMM and LVCSR systems is presented in Fig. 1, which shows performance of the two-session training condition on 3-, 10-, and 30-sec test segments from the 1998 evaluation [11]. The GMM system clearly outperforms the LVCSR system for the shortest test pieces, but the gap disappears by the time the test duration reaches 30 sec. It would be interesting to track performance as test duration continues to grow, but suitable data are not yet available. In the 1999 evaluation, variable-length test pieces were used,

ranging from close to 0 up to 60-sec pieces, averaging roughly 30 sec in length, with roughly 80% of the distribution lying between 15- and 45-sec durations. Not surprisingly, the performance of our systems on the variable-duration data closely tracked the pure 30-sec results from 1998, with too little representation at the extremes to yield meaningful time-stratified results.

1.3. The SNP System

The simple "bag of frames" approach of the GMM system works surprisingly well, but it ignores much useful linguistic information which should improve speaker recognition performance. In an attempt to capture some of this higher-level structure, in late 1997, Dragon began development on a new system which works from the same recognizer output as the basic LVCSR system described in Section 1.1. As we will show, the additional information provided by the sequential nonparametric (SNP) system, when combined with our existing systems, indeed yields a significant performance enhancement.

The SNP method addresses two shortcomings of the existing GMM and LVCSR systems:

– Independence of frames. Both the GMM and the LVCSR systems make little use of sequential information, i.e., how speech unfolds in time. The GMM system treats every frame as independent of all others (so that randomly permuting frames leaves the score unchanged) and the LVCSR system, though using somewhat more speech structure, still models frames within a single HMM state as independent.

– Dependence on an explicit parametric model. Modeling acoustic phenomena as a simple mixture of Gaussians in a very large dimensional space may be too crude to capture the fine structure of these events.

The SNP system [3] scores sequences of frames without making any parametric assumptions, by instead making direct comparisons to the target training data and computing a nearest-neighbor score. It works as follows:

The same forced alignments used by the LVCSR system are here used to identify sequences of frames in a test utterance corresponding to individual phonemes. Each such *phoneme token* is scored against all frame sequences corresponding to the same phoneme in the target speaker's training data. For each pair of tokens, the best alignment is determined via standard dynamic programming techniques, minimizing the Euclidean distance between aligned frames subject to various skipping and doubling penalties. To each test token, we assign the score corresponding to the best match (closest distance) to the target training data, normalized by the number of frames in the test token. Because some phoneme sequences in the test data can lack any reasonable match in the training data (due to data sparsity), and for general robustness, we then limit attention to the best scoring 75% of the tokens in the test utterance. We compute the sum of the best sequence comparison scores for this subset, divided by the total number of frames.

As always, appropriate score normalization is essential but, because the SNP system is nonparametric, there is no natural background model on which to

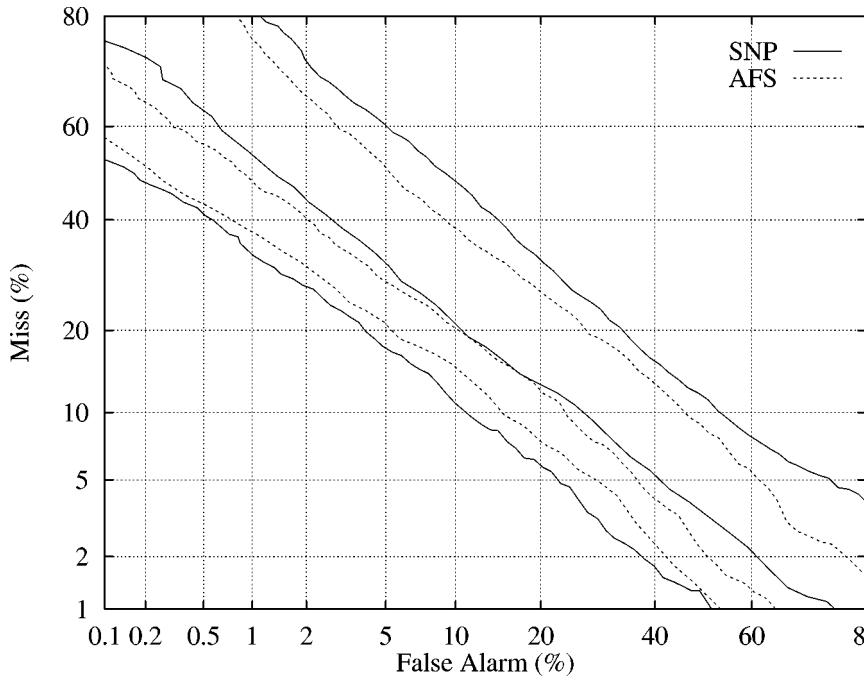

FIG. 2. Results on the female data set from the February 1997 NIST evaluation, comparing the SNP system to the AFS results, on the 3- (top pair), 10- (middle), and 30-sec (bottom) samples.

draw. Instead we use a cohort method, scoring the test data against 20 same-gender speakers and subtracting the average cohort score from the target score. We then perform the standard ZNORM across target speakers. (We have found no particular improvement using HNORM over ZNORM for the SNP system.) The cohort normalization makes the SNP system computationally much more expensive than either the LVCSR or the GMM system.

At the present time, the performance of the SNP system lags behind both the LVCSR and GMM systems. Nonetheless, although it is the least mature of our speaker ID systems, it already shows promise, adding valuable new information to the existing set. Performance on the 1999 NIST evaluation and the benefit gained by introducing the SNP system are illustrated in the next section.

Dragon is not the first site to explore nonparametric approaches to speaker recognition (see, e.g., [12, 13]), but most other approaches have treated frames as independent, neglecting sequential information. It is interesting to see how the SNP system compares to a simpler nonparametric system which, like the parametric GMM system, does not use sequential information nor LVCSR techniques. In such a system, instead of comparing frame sequences, we compare individual frames, assigning to each test frame the closest distance to any frame in the target training data. The best 75% of the resulting scores are then selected, averaged, and normalized as in the SNP system.

Figure 2 compares this all-frame scoring (AFS) system with the SNP system on data from the February 1997 evaluation, using the one-session training

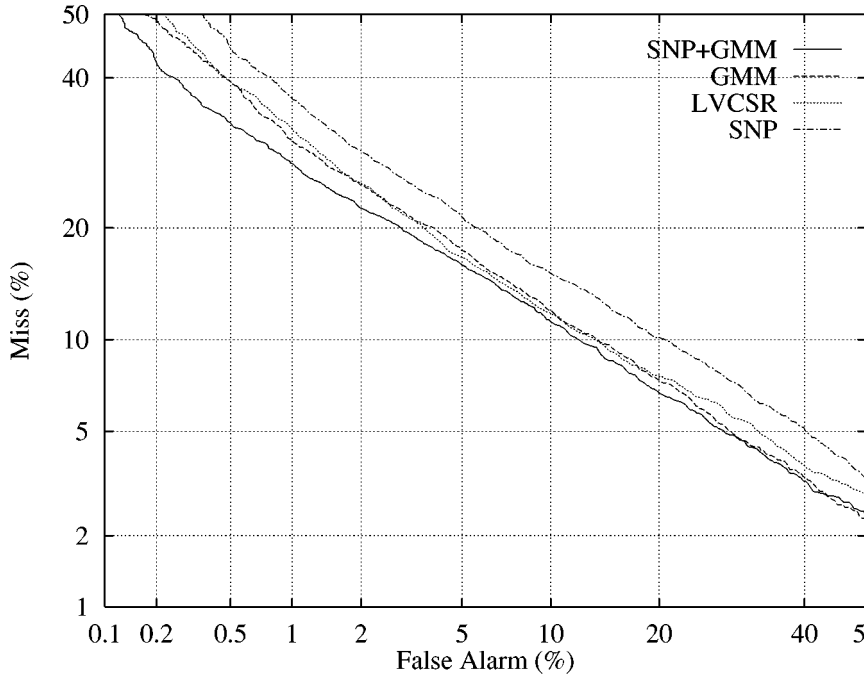

FIG. 3. Results on the combined April 1999 one-speaker evaluation data, including GMM (dashed), LVCSR (dotted), SNP (dot-dashed), and the combined SNP+GMM system (solid).

data. The results are remarkably similar to the GMM vs LVCSR comparison of Fig. 1. Here again, the LVCSR-based method is not as effective for short test segments but catches up to (and here surpasses) the simpler system when the test duration reaches 30 seconds.

1.4. Composite Systems

While the SNP system is not as effective as our older systems in isolation, it adds important new information to the mix. Figure 3 shows the performance on the 1999 NIST one-speaker evaluation for all three systems alone, as well as a system where the score for each test segment is the average score from the GMM and SNP systems. The latter composite system was submitted as Dragon's primary system in the 1999 NIST evaluation. We believe that this composite system gives the best performance to date on NIST's one-speaker task.

We have looked at a number of ways to combine the existing set of systems, ranging from simple linear interpolation to more sophisticated methods such as logistic regression. So far, we have seen no significant improvement over linear interpolation, with the best results coming from the SNP+GMM combination illustrated here. Ideally, we should seek to combine the systems based on the relative strengths of each, e.g., using test duration to weight simple systems more for short test segments, with the emphasis shifting to systems using more

detailed knowledge as the amount of data grows. This is a subject for future work.

1.5. Ongoing Research

While the results described above give some indication of the relative value of our three speaker ID systems at this time, they only begin to suggest the future potential of these techniques. Given the narrowing gap between frame-independent and LVCSR techniques as test duration grows, we would like to explore performance with longer test (and potentially, training) segments and with improved recognition. We believe that the current highly errorful recognition may well be limiting speaker recognition performance by forcing the LVCSR-based systems to score against incorrect phoneme labels and imperfect alignments. We hypothesize that the recognition problem may be especially acute for the shortest test pieces. Cheating experiments using correct transcripts are planned to gauge performance potential.

In addition, much work remains in improving the relatively new SNP system, including explorations of the best choice of "unit" for token comparison. For the purposes of the 1999 evaluation, we settled on the phoneme as a compromise between detailed structure and data sparsity, but other choices (triphones, multiphones, even whole words for commonly occurring instances) or variable-length units may provide better performance.

The LVCSR approaches provide a natural framework for incorporating higher-level knowledge than is available to the single-frame methods. We plan to examine the use of word choice, speaking style, and prosodic analysis in the near future. Of course, human subjects use such cues routinely when identifying speakers. And finally, we are examining ways to move from the single-speaker systems described above to analyzing data involving multiple speakers, as described in the next section.

2. TWO-SPEAKER TASKS

NIST recently introduced new components to their annual Speaker Recognition Evaluation: two-speaker detection and tracking. Instead of a sample of speech from a single speaker, we are provided with a 60-sec excerpt from a conversation between two speakers. The detection task is then to ascertain if a putative target speaker is present in the excerpt at all. The tracking task is to identify which frames, if any, contain speech from the putative speaker. The details of the two-speaker tasks and data sets appear in [14].

Eventually, we intend to apply all three systems used for the one-speaker task to the two-speaker problems. However, we decided to use our simplest method, the GMM, for the initial exploration of the two-speaker tasks. NIST provided two-speaker data in both the April 1999 evaluation and in a dry run held in September 1998; we have included results on both data sets in the sections that follow.

2.1. Two-Speaker Tasks: Baseline Performance

To establish a baseline to which we could compare potential improvements, we assembled a system based on the GMM method, using the same background and adapted speaker models as for the one-speaker task. In the following discussion we will only touch on the salient differences with respect to the one-speaker system.

For the baseline system, we run the channel normalization used for the one-speaker task over the entire two-speaker test sample as if it were speech from a single channel, effectively averaging the corrections for the two channels. This procedure clearly ignores issues specific to the two-speaker environment, such as overlapping speech and differences (gain, noise) between the two channels. We will return to these issues in later sections.

Once channel normalization is performed, we wish to identify the frames of the test sample which are likely to come from a target speaker. However, individual frame scores are too noisy to form the basis of a decision on speaker assignment. Our approach is to chop the conversation into segments which are relatively pure in a single speaker, while requiring segments to be at least a few hundred frames in length, to provide the needed statistical sample to separate target and background speech. There are a variety of ways to generate such segments; for the baseline system, we simply cut each conversation into regions bounded by silence, as defined by an energy threshold, and require at least 200 frames (2 sec) of data in each segment. We have found empirically that our performance degrades when much shorter segments are allowed.

We score individual frames as we did for the one-speaker task, computing a score difference between a generic background model and one adapted to the training data from the putative speaker. For each segment produced by the chopping algorithm, we compute the mean score difference over the frames above an energy threshold. For the tracking task, we assign each frame in a given segment the mean score difference over the segment. For the detection task, we select the best-scoring 50% of the segments and compute an average score difference over the frames in the selected segments.

We then apply ZNORM to normalize the scores from both the tracking and the detection tasks (for simplicity, since HNORM is greatly complicated by the two-speaker environment). The final results are shown in Fig. 4 for the April 1999 NIST evaluation data, where the detection task results are given by the solid curve and the tracking results by the dashed curve. Here we computed the means and variances required by ZNORM from two-speaker data from the September 1998 dry run. Also shown in Fig. 4 is a dotted curve indicating the performance of our GMM system on the one-speaker detection task for the April 1999 data.

Reducing the gap between one- and two-speaker performance will require more sophisticated approaches to the problems of channel normalization, speaker segmentation, and scoring in the two-speaker context. In the following sections we discuss a number of potential improvements to our baseline method.

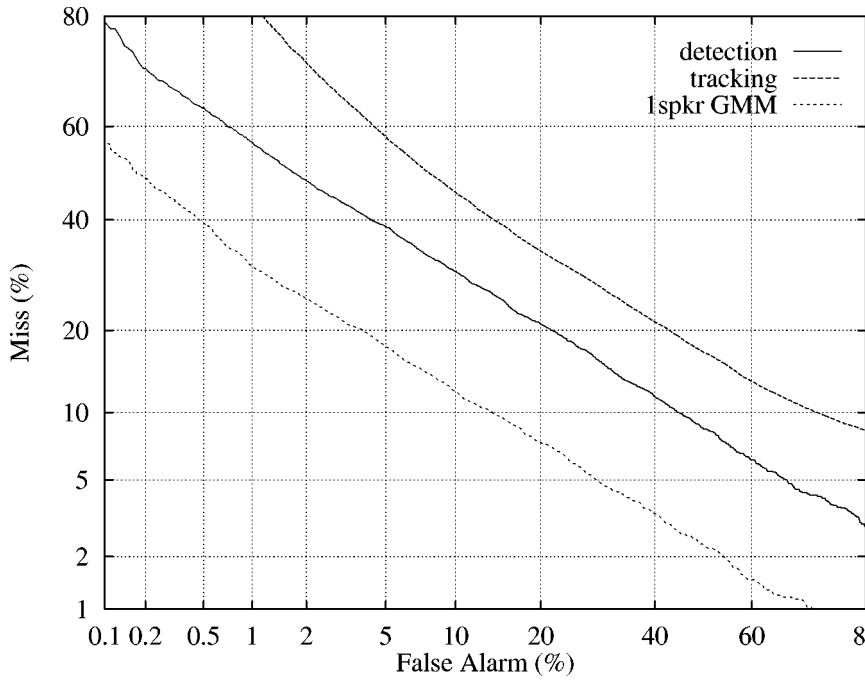

FIG. 4. Results on two-speaker detection (solid) and tracking (dashed) tasks for the April 1999 NIST evaluation. Also shown are results for the GMM system on the one-speaker task (dotted).

2.2. Alternative Methods for Segmentation

Segmentation of two-speaker conversations using silences is certainly a simplistic approach. It might seem that the use of more detailed information, such as speaker models, should improve our ability to distinguish speaker and background frames. We have investigated two methods of explicitly including the models into the segmentation task: sliding means and two-state HMM decoding. At this stage, neither of these approaches has outperformed silence chopping.

The sliding means approach replaces each individual frame's score with the mean score over a symmetric window around it, smoothing the noisy frame-by-frame scores. The size and shape of the averaging window can be manipulated to balance sensitivity to local information and score stability. A simple threshold can then be applied to the smoothed scores to assign each frame to speaker or background.

The decoding approach treats the tracking task as a single, two-state HMM problem. By using the speaker-adapted GMM as one state model and the background model as the other, we can "decode" two-speaker conversations into speaker and background frames. The two important parameters for this system are a bias term for the relative target/background probability and a switching penalty which affects the typical duration of each state.

We have implemented both the decoding and sliding means systems and have investigated a wide range of parameter settings for each. Representative results

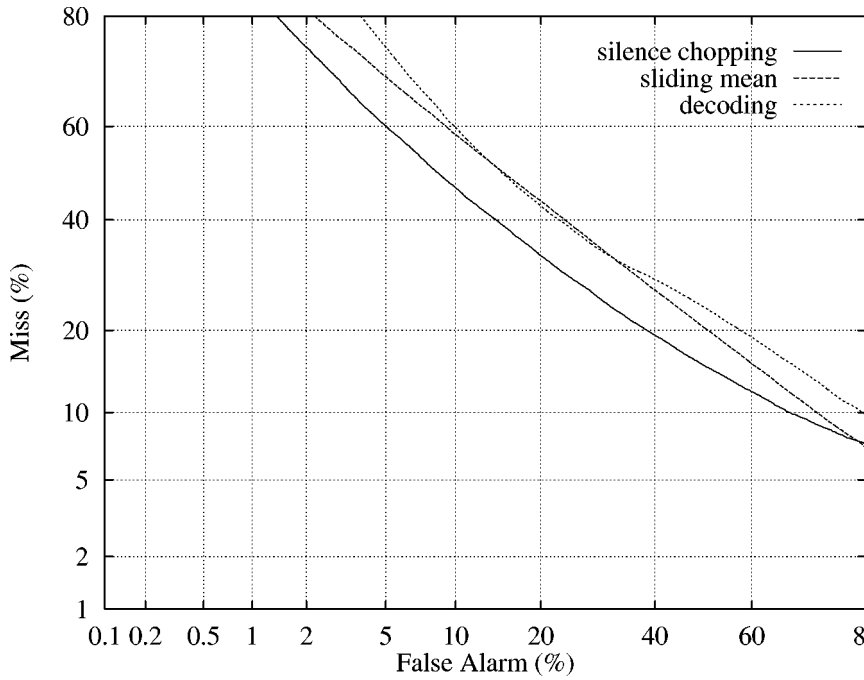

FIG. 5. Results for two-speaker tracking task using the sliding mean (dashed), decoding (dotted), and silence chopping (solid) methods, on September 1998 dry run data.

for these approaches on the tracking task of the September 1998 NIST dry run are compared in Fig. 5 to the baseline system of the previous section. It can be seen that the preliminary results for these more sophisticated approaches are not yet competitive with the naive silence chopping method.

We hypothesize that the poorer performance of the model-based methods is due at least in part to their greater sensitivity to inappropriate channel normalization and the presence of overlapping speech. The best results may ultimately come from adopting an intermediate approach to segmentation, which does not go so far as to construct a speaker-specific model, but which is specifically designed to search for speaker changes. Examples include Hotelling's T^2-test and the BIC criterion, as used for speaker change detection in Broadcast News recognizers [15]. We have done some initial explorations of these techniques, but so far have not found any significant improvements from their use over the silence chopping method.

Our lack of success with these explorations of segmentation has led us to seek a way to analyze comprehensively the problems specific to the two-speaker tasks.

2.3. "Cheating" Tests

The new problems presented by the two-speaker tasks can be broken down into the related areas of channel normalization, segmentation of the conversation into speaker turns, and selection of the segments corresponding

to a single speaker. We decided to use a series of "cheating" experiments to explore how best to address these issues. By "cheating", we are referring to the use of our knowledge of the true speakers in each conversation to allow perfect segmentation and segment selection. The details of which speaker is talking in which frames are provided by *locality* files, made available by NIST.

Our goal for the initial study was to create a test whose results could be directly compared to the one-speaker results. This was possible due to the way NIST designed the one- and two-speaker tasks: each one-speaker test utterance corresponded to one side of a two-speaker test conversation. However, the one-speaker tests were taken from the single-channel recordings so that effects such as overlapping speech or cross-channel noise were absent. An energy threshold was applied to the single-speaker recordings, and the segments corresponding to speech were concatenated together to produce the one-speaker test files. Using the locality files, we could create new versions of the one-speaker test files by extracting the corresponding segments from the two-speaker conversations. However, this version of the one-speaker tests would still contain the overlapping speech and channel noise from the other speaker in the conversation. We refer to this process as *perfect chopping*, i.e., identifying the frames containing speech from one side of a conversation, including any overlaps with the other side. Results on the concatenated two-speaker extracts can then be compared directly to our results on the standard one-speaker task.

We first considered the relative importance of chopping and channel normalization by constructing two tests:

(i) Perfect Chopping: in this test, we applied channel normalization to the two-speaker conversations as a whole, before chopping them using the locality files.

(ii) Perfect Chopping and Channel Normalization: here we first chopped the two-speaker conversations, and then applied channel normalization on the extracted segments. Absent an algorithm to subtract overlapping speech or noise, this test corresponds to the optimal two-speaker system.

Once we had constructed these pseudo-one-speaker tests, we processed them through our standard one-speaker GMM system. For simplicity, we restrict ourselves to the male tests in the discussion that follows, but the performance on the female test set is similar.

Figure 6 shows the results of these two cheating tests on the NIST September 1998 dry run test set, compared to the male one-speaker performance with the GMM system and the baseline results on the two-speaker detection task. Significant, and roughly equal, advances are made as we sequentially introduce perfect chopping (dotted) and perfect channel normalization (dashed). It can also be seen that, even for the optimal system, issues such as overlapping speech and channel noise from the other speaker continue to degrade performance in the two-speaker task relative to the one-speaker task.

The previous experiments have not addressed an important practical issue: even in the presence of perfect segmentation, it is a nontrivial task to select the segments corresponding to the same speaker. We have only established so far

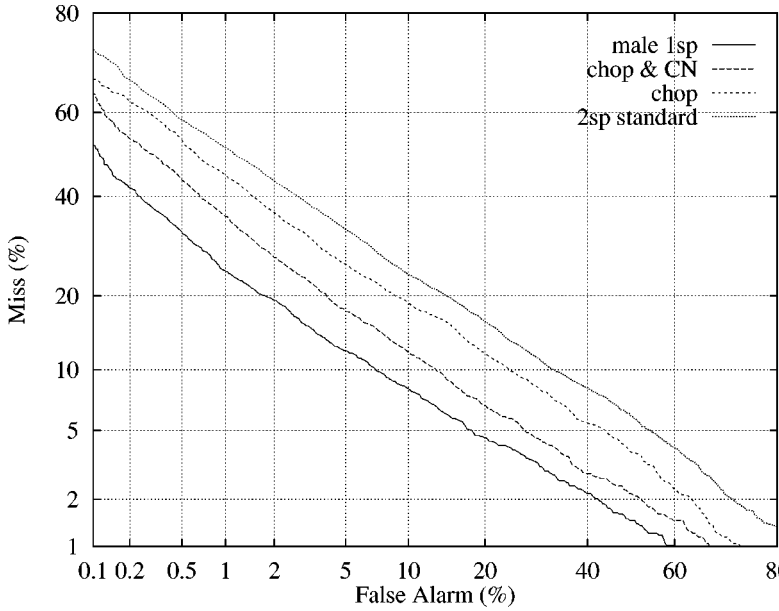

FIG. 6. Results of "cheating" tests on the NIST September 1998 dry run sample.

that together, optimal segmentation *and* segment selection yield a significant improvement in performance. As discussed above, our current approach to segment selection is simply to compute the mean *background–target* score for each segment produced by the chopping algorithm and to assign the best-scoring 50% of the segments to the putative speaker. However, there is a tension between the length of the segments and their associated purity. Clearly, we can ensure that each segment is very pure in a single speaker by chopping the conversation very finely. However, the resulting large variance in scores on these very short segments may make it impossible to choose the segments coming from the speaker of interest. Longer segments with lower purity may actually yield better performance, as the score variances can be made small enough to allow the algorithm to identify the correct segments more accurately.

We have examined the segment selection problem with a third cheating test, where we try to construct a best-case scenario: perfect determination of segment boundaries, and the appropriate channel normalization for the speaker of interest. For each two-speaker detection test, we start by chopping the conversation using the locality file information. The speaker of interest is defined as the target speaker, if actually present in the conversation, or otherwise either of the true speakers, chosen at random. We then compute our channel normalization based only on the segments coming from the speaker of interest, but apply the resulting corrections to *all* segments. The mean *background–target* score is calculated separately for all segments, and we choose the best scoring 50% of the segments to determine a speaker detection assignment, as done for the baseline system. In Fig. 7, we compare the performance of this system (dashed line) on the detection task to that of the

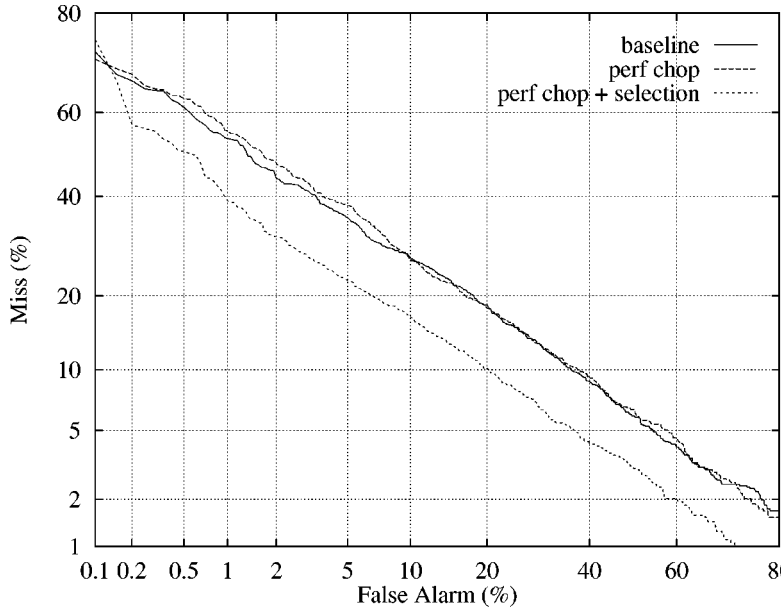

FIG. 7. Comparison of baseline (solid line) performance with cheating tests where the utterance is chopped according to perfect information but segments are selected according to their scores (dashed line), and with perfect chopping and segment selection (dotted line).

system using silence chopping (solid line), on a subset of the September 1998 dry run data (the tracking subset). The segmentation based on locality files actually performs arguably *worse* than the silence chopping. Thus, the silence chopping makes up for the impurities in its segmentation through the increased stability in the scores provided by the longer averaging windows. If we use the "perfect" segmentation, but select only the segments corresponding to the speaker of interest, the third (dotted) curve results, yielding performance comparable to that seen in the previous cheating test. Although we can see that the current chopping and segment selection is not optimal, it is also clear from these results that they will have to be optimized as an ensemble.

We have thus established that most of the difference between one- and two-speaker detection performance can be recovered through optimizing the related issues of segmentation, channel normalization, and segment selection. Ideally, one might hope for an integrated approach which simultaneously optimizes all three of these elements. A practical first step in this direction is to make the channel normalization more localized.

2.4. Two-Speaker Channel Normalization

Up to this point, we have been using the same channel normalization algorithm for the two-speaker tasks as we applied to the one-speaker problem. This algorithm computes a correction for each cepstral parameter based on the mean of that parameter over the whole utterance. To ensure that background noise or silence is not included in the average, we use only frames whose energy lies in the upper 50% of the range of energies observed in the utterance.

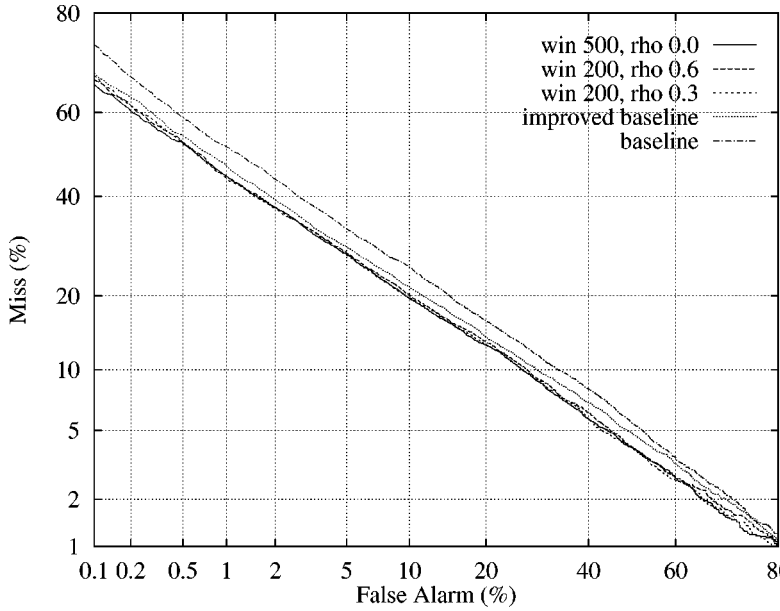

FIG. 8. Effect of sliding channel normalization for various choices of window size and ρ.

This provides robustness for a single-channel environment. However, in a two-channel environment, the channel characteristics can differ significantly. Discarding the lower 50% of the energy range can effectively suppress most of the frames from one side of the conversation. When we instead apply an absolute energy threshold, we obtain a significant increase in performance, as can be seen in Fig. 8 by comparing the baseline results on the September 1998 two-speaker detection task to the results of the simple energy threshold (labeled "improved baseline").

Further improvements in performance can be achieved by making the channel normalization more localized and thus more responsive to whichever channel is currently active. We have considered two approaches to constructing a more localized channel normalization algorithm. The first uses a sliding mean: for each of the cepstral parameters, we choose a target value, θ_i, and introduce a by-frame correction, Δ_i, of the form

$$\Delta_i = \theta_i - \left(\rho\mu_i^{glob} + (1-\rho)\mu_i^{loc}\right), \tag{1}$$

where μ_i^{loc} is the average of ith parameter over a predefined window centered at the current frame, μ_i^{glob} is the same average over the entire utterance, and ρ is a parameter defining the relative strength of the global and local corrections. An energy threshold is again applied to remove silence frames from both local and global means. We then use this channel normalization as input to our baseline system; the improved baseline system described above corresponds to $\rho = 1$. We have analyzed the September 1998 dry run sample with several different values of ρ and window size. The results appear in Fig. 8, compared to the original

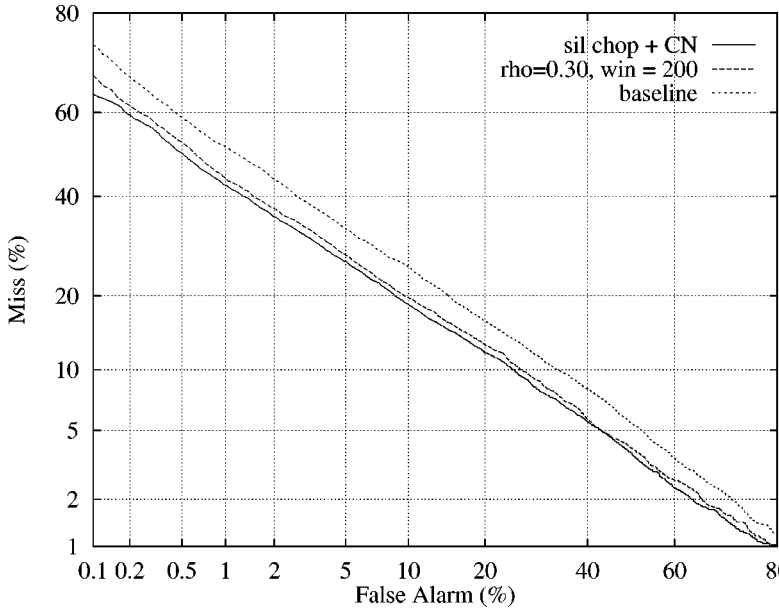

FIG. 9. Comparison of two-speaker detection results using the original GMM-based method (dotted), sliding channel normalization with $\rho = 0.30$ and a window of 200 frames (dashed), and channel normalizing within each segment produced by the silence chopping (solid). Results for the detection task on September 1998 dry run data.

and improved baseline systems applied to the same subset. Introducing some local information clearly improves the performance of the system, though the improvement does not seem to be a strong function of window size or ρ.

Another way to approach the channel normalization problem is to try to segment the conversation first, and then channel normalize the segments individually. If we can identify the speaker boundaries with some level of accuracy, this approach may provide higher speaker purities and thus yield better results than the more naive window-based approach. To this end, we have run the silence chopping first, yielding segments of at least 2 sec in length, and computed a flat ($\rho = 1$) channel normalization separately within each segment. Figure 9 compares the result of chopping first to the baseline approach, and a representative result from the sliding CN job ($\rho = 0.30$, 200-frame window), on the detection task for the September 1998 dry run data. Channel normalizing within segments from the chopping algorithm yields a small win over the sliding window approach.

2.5. Discussion

The results of Section 2.4 demonstrate that we can enhance performance by applying channel normalization locally within the segments of a two-speaker conversation. The cheating experiments have shown that even more can be gained if we can accurately identify a group of segments from a single speaker and channel normalize the ensemble. The interaction between the purity and the length of the segments produced by the chopping algorithm and our

corresponding ability to select segments coming from a single speaker will be a focus of our future studies. Ultimately, an approach which combines segmentation and clustering into a single process may provide the best results. Though we have seen some noticeable improvements in performance, it is clear that we are still far from the optimal system for the two-speaker task.

CONCLUSION

The speaker ID systems presented here offer a range of approaches to the speaker recognition problem and combine to yield state-of-the-art performance on the one-speaker task. Nonetheless, there remains significant room for improvement of single-speaker performance. Much is left to do to optimize the core systems individually, particularly the SNP approach. We also anticipate performance gains from exploration of the contexts in which each of the systems yields the best results, leading to improved methods of combining the systems.

The move from single-speaker to multispeaker data introduces a number of new challenges to speaker recognition systems, and we are only beginning to come to grips with the processes of speaker segmentation and segment selection. At this stage, we have explored these issues using only the GMM-based system, and there is clearly much potential for further refinements. We will also deploy our other (LVCSR-based) systems for the multispeaker tasks. In particular, we hope to take advantage of the access they provide to higher-level analysis, such as turn-taking cues for speaker separation. It seems likely that optimal performance will come from a combination of approaches.

REFERENCES

1. Peskin, B., *et al.*, Topic and speaker identification via large vocabulary continuous speech recognition. In *ARPA Workshop on Human Language Technology, Princeton, NJ,* 1993, pp. 119–124.
2. Newman, M., *et al.*, Speaker verification through large vocabulary continuous speech recognition. In *Proc. ICSLP-96, Philadelphia, PA,* 1996, pp. 2419–2422.
3. Corrada-Emmanuel, A., *et al.*, Progress in speaker recognition at Dragon Systems. In *Proc. ICSLP-98, Sydney, Australia,* 1998, pp. 1355–1358.
4. Peskin, B., *et al.*, Improvements in recognition of conversational telephone speech. In *Proc. ICASSP-99, Phoenix, AZ,* 1999, pp. 53–56.
5. Hunt, M. J., *et al.*, An investigation of PLP and IMELDA acoustic representations and of their potential for combination. In *Proc. ICASSP-91, Toronto, Canada,* 1991, pp. 881–884.
6. Reynolds, D. A., Experimental evaluation of features for robust speaker identification, *IEEE Trans. Speech Audio Process.* **2** (1994), 639–643.
7. Jelinek, F., *Statistical Methods for Speech Recognition.* The MIT Press, Cambridge, MA, 1997.
8. Orloff, J., *et al.*, Adaptation of acoustic models in large vocabulary speaker independent continuous speech recognition. In *Proc. ARPA Spoken Language Technology Workshop, Plainsboro, NJ,* 1994, pp. 119–122.
9. Reynolds, D. A., Comparison of background normalization methods for text-independent speaker verification. In *Proc. Eurospeech-97, Rhodes, Greece,* 1997, pp. 963–966.
10. Reynolds, D. A., Quatieri, T. F., and Dunn, R. B., Speaker verification using adapted Gaussian mixture models, *Digital Signal Process.* **10** (2000), 19–41.

11. Przybocki, M. A., and Martin, A. F., NIST Speaker Recognition Evaluations. In *Proc. LREC, Granada, Spain,* 1998, pp. 331–335.
12. Higgins, A. L., Bahler, L. G., and Porter, J. E., Voice identification using nearest-neighbor distance measure. In *Proc. ICASSP-93, Minneapolis, MN,* 1993, pp. II.375–II.378.
13. Bahler, L. G., Porter, J. E., and Higgins, A. L., Improved voice identification using a nearest neighbor distance measure. In *Proc. ICASSP-94, Adelaide, Australia,* 1994, pp. I.321–I.323.
14. Martin, A., and Przybocki, M., The NIST 1999 Speaker Recognition Evaluation—An overview, *Digital Signal Process.* **10** (2000), 1–18.
15. Wegmann, S., Zhan, P., and Gillick, L., Progress in Broadcast News transcription at Dragon Systems. In *Proc. ICASSP–99, Phoenix, AZ,* 1999, pp. 33–36.

FREDERICK WEBER received his undergraduate education at Grinnell College, obtaining B.A.s in physics and Russian in 1986. He performed his graduate work in experimental particle physics at the University of Wisconsin, receiving his Ph.D. in physics in 1993. He then engaged in five years of postdoctoral research on neutrino physics at CERN (Centre Europeen de Recherche Nucleaire), in Geneva, Switzerland, initially as a laboratory fellow, then continuing with postdoctoral appointments at UCLA and Harvard University. He joined Dragon Systems, Inc., as a research scientist in February 1999 and for the past year has had primary responsibility for Dragon's speaker ID systems.

BARBARA PESKIN, Principal Research Scientist at Dragon Systems, received her undergraduate degree in mathematics from Harvard University in 1975 and her Ph.D. in mathematics from the Massachusetts Institute of Technology in 1980. Before coming to Dragon, she taught at Mount Holyoke College, the University of Illinois, and Harvard University. Barbara joined the technical staff of Dragon Systems in 1991. Currently, she oversees Dragon's LVCSR effort in conversational telephone speech and has been especially involved in automatic transcription and in topic and speaker identification using the Switchboard corpus.

MICHAEL NEWMAN was an undergraduate at Cambridge University, receiving his degree in mathematics in 1987, before proceeding to Princeton University, where he completed his Ph.D. in theoretical physics in 1992. After two years of postdoctoral research at Harvard, he joined Dragon in 1994, where his first project was on language identification. Then followed several years of work on speaker identification, as well as the Switchboard automatic transcription task. Over the past two years, he has also been heavily involved in developing Dragon's commercial recognition products. He is currently a senior research scientist at Dragon.

ANDRÉS CORRADA-EMMANUEL received his A.B. in physics from Harvard University in 1981 and a Ph.D. in theoretical physics from the University of Massachusetts at Amherst in 1989. He has taught at Hamilton College, Swarthmore College, and Williams College. Prior to joining Dragon Systems, he did research on superfluids. Andres worked as a research scientist at Dragon from 1995 to 1998, where he was primarily involved in the creation of a Spanish language speech recognizer for conversational telephone speech and in speaker recognition research. He now runs a computer consultant business in Northampton, MA.

LARRY GILLICK is the Vice President of Research at Dragon Systems, where he has been conducting speech and language research since 1985. He received his B.A. from Swarthmore College in physics in 1972, his M.A. from Columbia University in 1974, and his Ph.D. in applied mathematics from MIT in 1980. Before joining Dragon, he taught and did research in statistics and in pattern recognition at MIT and at Northeastern University. At Dragon, Larry has done research and led projects in many different aspects of speech and language technology—rapid search, acoustic modeling, language modeling, confidence estimation, speaker ID, language ID, and topic detection and tracking.