A SEGMENT-BASED SPEAKER VERIFICATION SYSTEM USING SUMMIT[1]

Sridevi V. Sarma and Victor W. Zue

Spoken Language Systems Group
Laboratory for Computer Science
Massachusetts Institute of Technology
Cambridge, Massachusetts 02139 USA
{sree,zue}@sls.lcs.mit.edu

ABSTRACT

The main goal of this work is to develop a competitive segment-based speaker verification system that is computationally efficient. To achieve our goal, we modified SUMMIT [12] to suit our needs. The speech signal was first transformed into a hierarchical segment network using frame-based measurements. Next, acoustic models for 168 speakers were developed for a set of 6 broad phoneme classes. The models represented feature statistics with diagonal Gaussians, preceded by principle component analysis. The feature vector included segment-averaged MFCCs, plus three prosodic measurements: energy, fundamental frequency (F0), and duration. The size and content of the feature vector were determined through a greedy algorithm while optimizing overall speaker verification performance. We were able to achieve a performance of 2.74% equal error rate (EER) using cohorts during testing; and 1.59% EER using all speakers during testing. We reduced computation significantly through the use of a small number of features, a small number of phonetic models per speaker, few model parameters, and few competing speakers during testing (when cohorts are used).

1. INTRODUCTION

Speaker verification involves the task of automatically verifying a person's identity by his/her speech through the use of a computer. The outcome of speaker verification is a binary decision as to whether or not the incoming voice belongs to the purported speaker. Speaker verification has been pursued actively by researchers, because it is presently a palpable task with many uses that involve security access authorizations. In the past, applications for speaker verification systems mainly involved physical access control, automatic telephone transaction control (e.g., bank-by-phone), and computer data access control. However, due to the revolution in telecommunications, uses for speaker verification systems also include Internet access control, and cellular telephone authorizations.

Figure 1 illustrates the basic components of a speaker verification system. The feature extraction component determines acoustic measurements from the user's speech signal that are relevant to inter-speaker differences. During training, the acoustic features are used to build speaker-specific models. During testing, measurements extracted from the test data are scored against the stored speaker models to see how well the test data match the reference models. The speaker is accepted or rejected based on this score. Many details are left out of the block diagram, such as the type of text the system prompts, the features the system extracts, and the speaker models and classifiers the system implements. For a detailed tutorial on speaker verification, refer to [7].

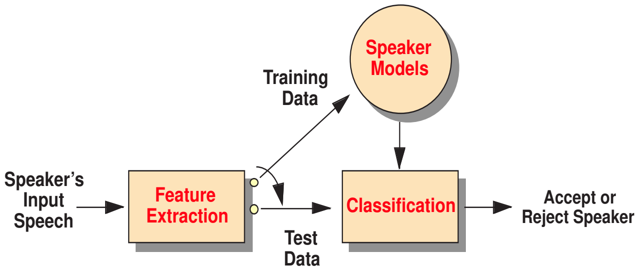

Figure 1: General speaker verification system

In this paper, we describe how we developed a speaker verification system that extracts acoustic features from speech segments. Our investigation is motivated by past observations that speaker-specific cues may manifest themselves differently depending on the manner of articulation of the phonemes [10]. By treating the speech signal as a concatenation of phone-sized units (segments), one may be able to capitalize on measurements for such units more readily. A potential side benefit of such an approach is that one may be able to achieve good performance with unit (i.e., phonetic inventory) and feature sizes that are smaller than what would normally be required for a frame-based system, thus deriving the benefit of reduced computation.

2. SYSTEM DESCRIPTION

2.1. Corpus

The TIMIT [4] corpus was used in our experiments. TIMIT consists of 630 speakers, 70% male and 30% female, representing 8 major dialect regions of the United States. We selected a subset of 168 speakers from the standard NIST-train set for development, and a separate set of 168 speakers (TIMIT's standard NIST-test and NIST-dev sets) for evaluation. Eight sentences (SX, SI) were used to develop each speaker model, and the remaining 2 SA sentences were used to test each speaker. We clustered the 61 TIMIT-labeled phones into 6 broad manner classes (vowels, weak fricatives, strong fricatives, nasals, stops,

[1] S. Sarma receives support from the National Science Foundation. This research is also supported by a contract from Bell South Intelliventures.

silence), in the hope of developing robust models that can distinguish among speakers.

2.2. Signal Representation

Past observations have shown that the Mel-frequency-based cepstral coefficients (MFCCs) and prosodic features are useful for speaker verification [6, 11]. Thus, the initial set of features used consisted of MFCCs and several prosodic measurements. Specifically, 14 MFCCs, the logarithm of energy, duration and fundamental frequency (FO) were computed[2]. The features were averaged across speech segments, which were proposed by a segmentation algorithm implemented in SUMMIT.

2.3. Feature Search

The 17 measurements mentioned above represent a pool of possible features to characterize segments. We did not use all 17 measurements in the system for several reasons. First, some features may discriminate between speakers well, while others may not. Second, some of the measurements may be correlated or essentially carrying the same information. In addition, training models with high dimensionality may be problematic since not much data are available per speaker. Finally, computation increases as the number of features increases, which may become prohibitive if all 17 measurements are used in the system.

To find a (sub)-optimal subset of the 17 features, we conducted a greedy search [2]. At every decision point in a greedy algorithm, the best choice, based on an optimality criterion, is selected. Our criterion is the speaker verification performance of each proposed feature set. Performance is measured in terms of a distance metric that minimizes the two types of errors, false rejection of true users (FR) and the false acceptance of impostors (FA). While our goal is to minimize both types of errors, we have chosen to weigh the cost of false acceptances of impostors more than the cost of false rejections of true users. Specifically, we obtain the receiver operating characteristic (ROC) curve (FR vs. FA) for each feature set. The system's performance is then measured in terms of a distance between the point on the feature's ROC curve that corresponds to the lowest false acceptance rate, to the origin, which corresponds to the ideal performance of 0% error.

The search algorithm begins by obtaining FR rates and FA rates for the speaker set, using each of the 17 features. Thus, we obtain 17 performance results corresponding to each measurement. The feature that results in the smallest distance measure (minimum error rates) is chosen as the best 1-dimensional measurement. Next, the best 1-dimensional feature is combined with each of the remaining measurements. Two-dimensional feature sets are grouped in this fashion, and are each used to test the speakers. The best 2-dimensional feature vector, in terms of speaker verification performance, is then used for the next stage of the search. The search continues to accumulate dimensions to the feature set until there is no longer significant improvement in speaker verification performance, or until performance actually degrades as more features are added.

[2]To estimate F0, we used the ESPS tracker, in particular the FORMANT function [3]. Although the tracker estimates probabilities of voicing for each frame, we retained F0 information for every frame, regardless of whether the underlying sounds were voiced or unvoiced.

2.4. Training and Testing

In order to train and test the utterances, each utterance must first be delineated into segments that correspond to the broad manner classes. In our case, this is accomplished through a forced-alignment of the signal with the underlying phonetic transcription, after the phone labels have been collapsed into their corresponding broad classes. To facilitate this alignment, we must first develop a set of speaker-independent (SI) models. For our experiments, we actually developed two sets of SI models, one to test on the development set and another to evaluate on the test set speakers.

Diagonal Gaussian speaker models of the segment-based acoustic features are then developed using the forced transcriptions for each speaker and broad class. Conventional maximum likelihood estimates are used to approximate the distribution parameters. To ensure that the features fit the diagonal models better, principal components analysis (PCA) is performed on the acoustic features before developing the models. In our experiments, we did not reduce dimensionality with PCA since the feature search already prunes the number of features used in the system.

Once speaker models are developed, test utterances are scored against these models to classify speakers and make verification decisions. It is ideal to compare test utterances to all speaker models in the system, and accept the purported speaker if his/her model scores test data the best. However, computation becomes more expensive as speakers are added to the system. Since speaker verification is simply a binary decision of accepting or rejecting a purported speaker, the task should be independent of the user population size. To keep our system independent of the number of users and computationally efficient, we implemented a technique called *cohort normalization*. For each speaker, we pre-detected a small set of speakers, called a cohort set, who are acoustically similar to the purported speaker. For each feature set and speaker, we found 14 nearest neighbors using the Mahalanobis distance metric.

To accept or reject a speaker, we compute forced alignment scores, described in [9], for the purported speaker's two test utterances. The scores are computed using 15 models, the speaker's model and his/her 14 cohort models. These scores are then sorted, and the speaker is accepted if the score using his/her model is in the top N scores of the 15 results. N is a rank-threshold that we varied.

3. FEATURE SELECTION

3.1. Development Set

We conducted a feature search using 168 development speakers. After the first stage of the search, 10 out of the 17 features were eliminated due to significantly poorer performances. We realized that such pruning will result in a search that is not greedy in the strictest sense of the word. Eventually, the algorithm led us to obtain a (sub)-optimal 6-dimensional feature set which consisted of energy, MFCC10, MFCC8, MFCC4, MFCC12, and MFCC6. The feature search terminated because performance leveled off and degraded after the sixth stage.

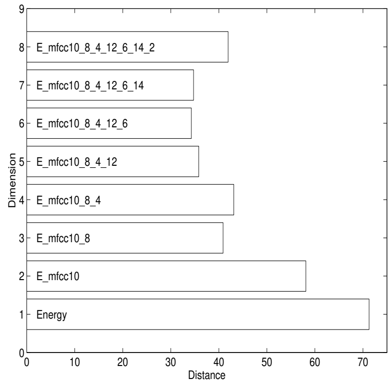

Figure 2: Distances for best feature sets of each search stage using development speaker set

3.2. Test Set

To investigate whether the feature selection process is independent of the speakers used, we conducted an identical search using a set of 168 test speakers. The optimal feature set found in this search also consisted of 6 dimensions: energy, MFCC10, MFCC5, MFCC8, MFCC6, and MFCC14. The feature search terminated after performance degraded using 7, 8-dimensional feature sets. The performances of the best stages of each search stage are shown in Figures 2 and 3.

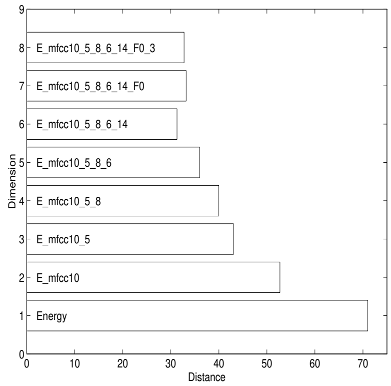

Figure 3: Distances for best feature sets of each search stage using test set

3.3. Discussion

As illustrated in Figures 2 and 3, speaker verification performance initially improves as more measurements are added to the feature set (distance decreases), because the additional features contribute further speaker-specific information. Also, there are sufficient amounts of training data to accurately estimate the speaker model parameters. However, adding features eventually degrades performance, presumably because not enough training data is available to accurately estimate the model parameters.

The features selected for each speaker set vary slightly (4 out of 6 features are the same). The differences may be due to the fact that the SI models used to create forced transcriptions for each speaker set are different. However, general trends in each search are similar. In the first stages of both searches, we found that energy performed well and duration performed poorly. To analyze these results, we computed the average Mahalanobis distances between cohort models for energy and duration. We found that the average Mahalanobis distance for the energy models is approximately twice that of the duration models, suggesting that the energy speaker models are much more different than the duration speaker models within cohort sets. Consequently, it is easier to distinguish between speakers using energy as a feature than using duration as a feature.

4. SPEAKER VERIFICATION PERFORMANCE

4.1. Effects of Cohort Normalization

As previously mentioned, computation during testing is reduced by only scoring test data against the purported speaker's model and models of the his/her cohort set. This technique is based on the assumption that speaker models outside of the cohort set will not adversely affect speaker verification performance. Since these outliers are considered too different from the purported speaker, their models are expected to match the test data poorly compared to the speaker models within the cohort set. If this is the case, the ROC curves corresponding to performance using all speakers during testing can be obtained from the ROC curves using only cohort sets during testing, via normalization. The normalization divides the number of false acceptances obtained for a feature set, using for each speaker only the 14 speakers in his/her cohort set as impostors, by the number of possible false acceptances when all the remaining 167 speakers pose as impostors for each speaker (168 speakers x 167 impostors).

We first normalized the results for the (sub)-optimal feature sets found after feature selection. Then, we verified whether these normalized approximations are reasonably close to the performances using all speakers during testing by repeating the experiments on the feature sets using all speakers during testing. As Table 1 illustrates, the normalized (Cohorts) equal error rates, the rates at which two possible errors are equal, are similar to the results obtained using all speakers during testing (No Cohorts). The performances do not match exactly and causes of the discrepancies could be due to at least two reasons. First, we selected cohorts using the Mahalanobis distance, whereas during testing we compared speakers using forced-paths scores. Second, the cohorts were not selected in a manner that maximized a spread around each speaker as they were in [8]. A spread prevents impostors that are far from the purported speaker, but even further from the purported speaker's cohorts, to be falsely accepted.

Speaker Set	Feature Selection	**Old** SUMMIT		**New** SUMMIT	
		Cohorts	**No Cohorts**	**Cohorts**	**No Cohorts**
Dev	energy_mfcc10_8_4_12_6	5.43%	6.27%	3.54%	1.19%
Test	energy_mfcc10_5_8_6_14	4.13%	4.00%	2.74%	1.59%
Test	17 features	6.65%	15.15%	–	–

Table 1: Performance: All results are in terms of EER.

4.2. Effects of New SUMMIT Recognizer

In addition to observing the effects of using cohorts during testing, we determined whether a recent improvement in the probabilistic framework for acoustic modeling in SUMMIT affected speaker verification performance[3]. To observe how performances of the selected feature sets are affected, we evaluated the 168 development and test speakers using the improved SUMMIT system. As Table 1 illustrates, the new recognizer significantly improves speaker verification performance over the old recognizer.

4.3. Effects of Feature Search

To observe whether the performance of the (sub)-optimal features sets found above improves over using all 17 features in the system, we evaluated the 168 test speakers using the 17 initial measurements. The results are also shown in Table 1. As expected, performance was poor. Presumably, not enough data were available to accurately estimate the large number of model parameters. Consequently, when performance using all speakers during testing is poor (>10% EER), normalized cohort approximations are inaccurate.

4.4. Performance Comparison

Below, we compare our system with two other state-of-the-art systems that also use the TIMIT corpus. One system implements HMMs to represent speakers [6], while the other uses neural networks [1]. Both systems, if tested using the same decision algorithm as ours, reach the ideal performance of 0% error.

Unlike the HMM and neural network (NN) system, our system does not achieve perfect performance. However, performance degradation is somewhat compensated by computational efficiency. We designed a simple system and reduced computation in a variety of ways. First, we used only 6 acoustic features, as opposed to 16 or 32 (NN and HMM respectively), to represent the speech signals. Second, we developed speaker models of 6 broad phonetic classes, as opposed to 31 as in the HMM system. Third, each of the 6 broad classes is represented by a single diagonal Gaussian distribution, as opposed to mixtures of Gaussians (as in HMM system) or the nonlinear distributions that neural networks typically produce. The two latter models have more parameters to estimate, and hence require more computation during training. Finally, we reduce computation during testing by using only a set of speaker models similar to the purported speaker's model, as opposed to using all the speaker models in the system.

[3] SUMMIT's recent modification and improvement is discussed in [5].

Computation, in terms of the number of training parameters, is approximated on the order of 10^6 for the HMM system and on the order of 10^4 for the SUMMIT speaker verification system. Not enough information is reported on the neural network system to reliably estimate the number of training parameters.

5. CONCLUSIONS

As described above, our system achieves a performance of 1.59% EER when all speakers are used during testing. In the process, we significantly reduced computation in many ways. We believe that by considering the speech signal as a concatenation of phone-sized units, we capitalized on measurements for such units more readily.

Future work includes representing acoustic features with more complex distributions, adapting speaker models [6], and conducting feature searches on NTIMIT and CTIMIT to find robust features for the telephone and cellular telephone domains, respectively.

6. REFERENCES

[1] Bennani, Y., "Speaker Identification Through Modular Connectionist Architecture: Evaluation on the TIMIT database," *Proceedings of ICSLP*: 607-610, 1992.

[2] Cormen, T., Leiserson, C., Rivest, R., "Introduction to Algorithms," *The M.I.T. Press* : 1990

[3] Doddington, G., Secrest, B., "An integrated pitch tracking algorithm for speech systems," *Proceedings of ICASSP*: 1352-1355, 1983.

[4] Garofolo, J. et al, "DARPA TIMIT Acoustic-Phonetic Continuous Speech Corpus CD-ROM," *National Institute of Standards and Technology*: 1990.

[5] Glass, J., Chang, J., McCandless, M., "A Probabilistic Framework for Feature-Based Speech Recognition," *Proceedings of the ICSLP*: 1996

[6] Lamel, L., Gauvain, J., "A Phone-based Approach to Non-linguistic Speech Feature Identification," *Computer Speech and Language*: 87-103, 1995.

[7] Naik, J., "Speaker Verification: A Tutorial," *IEEE Communications Magazine*: 42-47, 1990.

[8] Reynolds, D., "Speaker identification and verification using Gaussian mixture speaker models," *Speech Communication*: 91-108, 1995.

[9] Sarma, S., "A Segment Based Speaker Verification System Using SUMMIT," *M.I.T. SM Thesis, Department of Electrical Engineering and Computer Science*: 1997.

[10] Wolf, J., "Acoustic Measurements for Speaker Recognition," *M.I.T PhD Thesis, Department of Electrical Engineering and Computer Science* : 1969.

[11] Yegnanarayana, B., Wagh, S., Rajendra, S., "A speaker verification system using prosodic features," *Proceedings of ICASSP*: 1867-1870, 1994,

[12] Zue, V., Glass, J., Phillips, M., Seneff, S. "The SUMMIT Speech Recognition System: Phonological Modeling and Lexical Access," *Proceedings of ICASSP*: 49-52, 1990.