IBM MODEL-BASED AND FRAME-BY-FRAME SPEAKER-RECOGNITION

Homayoon S. M. Beigi, Stéphane H. Maes, Upendra V. Chaudhari, and Jeffrey S. Sorensen

Human Language Technologies Group
IBM Research, T.J. Watson Center
P.O. Box 218, Yorktown Heights, NY 10598
EMail: beigi@watson.ibm.com

ABSTRACT

Amidst the recent crazes for emerging technologies like speech recognition and biometrics, speaker recognition is slowly reaching the maturity to be deemed practical in many different applications. This paper presents new approaches for text-independent speaker recognition.

The performances of the model-based algorithm presented concurrently at the ICASSP'98 conference and the frame-based algorithm presented in this paper are compared here.

The engine, described here, provides multiple functionalities including those of identification, verification and classification. The modes of operation and design choices allow for tight integration of the speech recognition and speaker recognition engines in a broad sense. This new architecture as well as the results obtained for very specific tasks undoubtedly announce myriads of new applications where both technologies complement each other and can no longer be clearly distinguished as illustrated by the concept of speech biometrics. Hands-free and eyes-free human/machine transactions are moving a step further toward easier and more efficient interfaces and speech transactions are becoming more ubiquitous.

RÉSUMÉ

Au sein du récent engouement pour de nouvelles technologies comme la reconnaissance de la parole et les biométriques, la reconnaissance du locuteur atteint une maturité permettant d'envisager des réalisations pratiques. Cet article présente de nouvelles techniques de reconnaissance du locuteur quel que soit ce qu'il dit.

Nous comparons les performances des différents algorithmes proposés lors de la conférence ICASSP 1998 avec l'approche trame par trame presenteé dans ce papier. Le moteur présenté fournit différentes fonctions: identification, vérification et classification. Les choix de modalités et de conception permettent une étroite intégration de la reconnaissance de la parole prise au sens large et de la reconnaissance du locuteur. Cette nouvelle architecture de même que les résultats obtenus pour des tâches bien spécifiques annoncent une myriade de nouvelles applications où ces deux technologies se complémentent au point de ne plus être distingués. Les interactions homme/machines en mode "mains libres et yeux libres" font un grand pas dans la directions d'interfaces plus aisés et plus efficaces. La parole devient davantage le vecteur préferré de d'interaction.

1. INTRODUCTION

This paper discusses new developments in the field of Speaker Recognition. In particular, it describe the algorithms and implementations of our speaker recognition engine. The recognition engine entails two different implementations, a model-based approach and a frame-by-frame approach, which will be compared here. The merits and short-comings of each approach are discussed and results are presented on clean and noisy data obtained in a relevant environment.

These algorithms and their implementation may be deployed for application domains such as the desktop, a client-server environment, an embedded system or the telephony environment.

We have focussed our efforts on solving the text-independent and language-independent case rather than text-dependent, text-prompted or other variations of the problem. Indeed, only a text-independent engine can remain transparent to the user throughout a voice transaction.

Similarly, in order to maintain the non-obtrusive aspect of our system, the speaker recognition engine is fully integrated with the IBM speech recognition engine: a speaker-independent version of ViaVoice^{TM}.

The integration of a speech recognition system with text-independent speaker recognition opens an array of new possibilities, including high-security access control.

The overall architecture is client-server based, with a light acoustic front-end client. The speech and speaker recognition engines reside on the server. With bit-rates lower than 4Kb/s, the real-time commands given to a light-weight client may be transmitted to a remote server via wire-less modems.

The engine is SVAPI (Speaker Verification API) compliant [1]. In this context, we define, in detail, the different functions that it offers: Enrollment, Speaker Verification, Speaker Identification, Speaker Classification and Speech biometrics. Examples of applications using these different modalities are cited.

The following two sections describe the different techniques which have been implemented in the IBM Speaker Recognition system. The first technique is based on computing the distance between a trained model and a test model for individual speakers. Second, a frame-by-frame technique is presented which enables immediate processing of the speech signal for obtaining a recognition result. Then, results are presented on the performance of these two techniques. These results are obtained on a data set recorded privately, with and without noise. The merits and shortcomings of both techniques are compared and a conclusion is given toward the end.

2. THE MODEL-BASED APPROACH

The model-based approach is used to train the system in a manner which will be used by both model-based and frame-by-frame techniques. This section describes the training and recognition phases of the system using these models. As it was mentioned earlier, the speaker recognition system described here includes Speaker Verification, Speaker Identification and Speaker Classification.

The first thing which should be achieved is the creation of training models for the population of speakers in the database. This is done by computing a model $\mathcal{M}_i$ for the i^{th} speaker based on a sequence of M frames of speech, with the d-dimensional feature vector, $\{\vec{f}_m\}_{m=1,\ldots,M}$. These models are stored in terms of their statistical parameters, such as, $\{\vec{\mu}_{i,j}, \mathbf{\Sigma}_{i,j}, \vec{C}_{i,j}\}_{j=1,\ldots,n_i}$, consisting of the Mean vector, the Covariance matrix, and the Counts, for the case when a Gaussian distribution is selected. Each speaker, i, may end up with a model consisting of n_i distributions. Now, if a distance measure were available for comparing two such models, we would be able to devise a speaker recognition system with many different capabilities including Speaker Identification, Speaker Verification and eventually Speaker Clustering by creating a hierarchical structure. Such distance measure has been proposed by the authors in the concurrent ICASSP'98 conference of reference [2].

The training data is stored using a hierarchical structure so that accessing the models would be optimized at the time of recognition. The Speaker Verification is implemented by extracting a set of speakers (with their models) from the training database considering only those speakers with close proximity, as given by the distance measure of [2], to the speaker with the claimed ID. The claimant's sample speech is then used to generate a test model which is compared to the models in the Cohort set. The models are sorted by the distance and the training model with the smallest distance from the test model is used to obtain the verification result. In this operation, a background model is also added to the set of trained models to establish a rejection mechanism. If the background model or any speaker other than the claimant comes up at the top of the sorted list, the claim is rejected. Otherwise, it is accepted.

In case of Speaker Identification, the claimant's test model may be compared, using the same distance measure, to all the models in the database including that of the background model. Based on the result of this comparison, the label of the model with the smallest distance to the test model is returned. Please note that this may be a rejection if the background model is the closest model to the test model.

Speaker Classification may be done by any hierarchical structure devised on the inter-model distances given by the distance measure of reference [2]. This classification is very useful in many different occasions including the narrow-down of the search space (models to be explored) for doing Speaker Recognition. The speaker segmentation system presented in [3] uses such classification technique by utilizing the method presented in this section.

The distance measure as defined by reference [2] is devised such that it may compute an acceptable distance between two models with different number of distributions n_i. The advent of a method for comparing two speakers solely based on their models, gives us the advantage of not having to carry the features around. Using this scenario, as mentioned earlier in this section, only the statistical parameters are recorded. This also makes the job of comparing two speaker much less computationally intensive. The reason for the computational simplification is the parametric representation of each speaker and the smaller number of computations needed to compare two models based on their parameters versus using the actual features to do this comparison as presented in the following section. A short-coming of using this distance measure for the recognition stage, however, is that the engine would have to wait until all the speech is collected from the test individual (claimant) to be able to start its computation. The method described in the next section alleviates this problem; however, it requires the actual features for computing the distance measure.

3. THE FRAME-BY-FRAME APPROACH

Let $\mathcal{M}_i$ be the model corresponding to the i^{th} enrolled speaker. $\mathcal{M}_i$ is entirely defined by the parameter set, $\{\vec{\mu}_{i,j}, \mathbf{\Sigma}_{i,j}, p_{i,j}\}_{j=1,\ldots,n_i}$, consisting of the mean vector, covariance matrix, and mixture weight for each of the

n_i components of speaker i's Guassian Mixture Model (GMM). These models are created using training data consisting of a sequence of M frames of speech, with the d-dimensional feature vector, $\{\vec{f}_m\}_{m=1,\ldots,M}$, as described in the previous section. If the size of our population is N_p, then the set of models we choose from, our model universe, is $\{\mathcal{M}_i\}_{i=1,\ldots,N_p}$. Whether the context is verification or identification, the fundamental goal is to find the i such that $\mathcal{M}_i$ best explains the test data, represented as a sequence of N frames, $\{\vec{f}_n\}_{n=1,\ldots,N}$, or to make a decision that none of the models describes the data adequately. We use the following frame-based weighted likelihood distance measure, $d_{i,n}$, in making the decision:

$$d_{i,n} = -log\left[\sum_{j=1}^{n_i} p_{i,j}p(\vec{f}_n|j^{th} \text{ component } of \mathcal{M}_i)\right], \quad (1)$$

where, using a Normal representation,

$$p(\vec{f}_n|\cdot) = \frac{1}{(2\pi)^{d/2}|\Sigma_{i,j}|^{1/2}} e^{-\frac{1}{2}(\vec{f}_n-\vec{\mu}_{i,j})^t\Sigma_{i,j}^{-1}(\vec{f}_n-\vec{\mu}_{i,j})} \quad (2)$$

The total distance, D_i, of model $\mathcal{M}_i$ from the test data is then taken to be the sum of all the distances over the total number of test frames.

$$D_i = \sum_{n=1}^{N} d_{i,n}. \quad (3)$$

For verification the the set of models considered are those which belong to the members of a predetermined cohort of the claimant augmented by a variety of background models. Using this set as our model universe, the test data is verified if the claimant is the one whose model has the smallest distance; otherwise, it is rejected.

For identification, no cohort structure is used. Simply, the model with the smallest distance is chosen. By comparing the smallest distance to that of a background model, we could provide a method to indicate that none of the original models match very well. Alternatively, a voting technique may be used for computing the total distance.

As mentioned in the previous section, the distance measure given by this section may be computed as the frames of speech are being produced by the test speaker. However, since the data would have to be kept around for computing the distances among the speakers, this distance measure is not being used for conducting the training. The training is being done, as mentioned in the beginning of this section, by the model-based technique of the previous section.

4. IMPLEMENTATION ISSUES

This section presents the implementation details of the engine and the test data. The features used in the Speaker Recognition engine do not contain any dynamic information in the present implementation.

The **enrollment** is done by using about 20 seconds of speech from each speaker in the training set and storing the associated models as discussed earlier. In storing these models, a hierarchical clustering is done to speed up the recognition stage. For **verification**, 2 and 4 seconds of data were used in two different tests run with both the model-based and frame-by-frame approaches. Results of these test are given in the next section. There were 60 speakers present in the database. The original data was collected in a clean environment. Then, a real, additive noise with the signal to noise ratio of $19dB$ was added to the clean speech data. This noise was collected on the floor of the COMDEX computer show. The results presented in the following section show the performance of the system when applied to the data with and without the addition of this noise. The cohort is obtained from the hierarchical structure which is built on the speaker database. Once clustered, in this structure, the classification of the speakers is done by traversing it and finding close speakers [2, 4, 3].

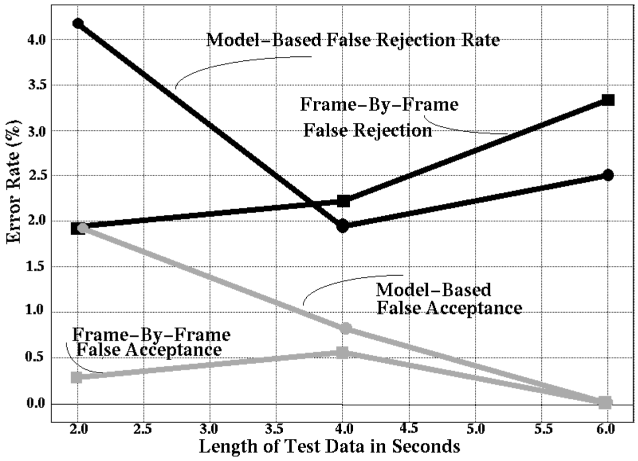

Figure 1: Speaker Verification using Clean Data

5. RESULTS

Speaker Verification using the model-based and frame-by-frame approaches has been tested with the above data. Figures 1 and 2, show the verification results for the clean and noisy data respectively. The system was trained using the model-based approach with about 20 seconds of data once with and once without the addition of noise to the data. Then, the trained models were stored using a hierarchical structure based on the closeness of the models to one-another. The distance measure given by reference [2] was used for clustering the speakers. At the event of verification to obtain the False Rejection results, the speaker's correct ID was claimed and it was either rejected or accepted. 6 tests were done for each speaker, amounting to 360 tests for each scenario. The false rejection rate was noted. Then, the same data was presented, this time using a false ID, randomly selected from the other 59 IDs not including the correct speaker. The results were noted as the False Acceptance rate of Verification. This was repeated once with 2 seconds of test data and once with 4 seconds of test data. The test was also conducted once with noisy data and once with clean data, using both model-based and frame-by-frame approaches. The speed of operation of the model-based verification was about 2 to 5 times that of the frame-by-frame approach. Of course, in these tests, due to their batch nature, the speeds are simply compared with each other. However, in a real test, since the frame-by-frame approach may start processing the data as it is being generated, it will not fall behind from the model-based counterpart.

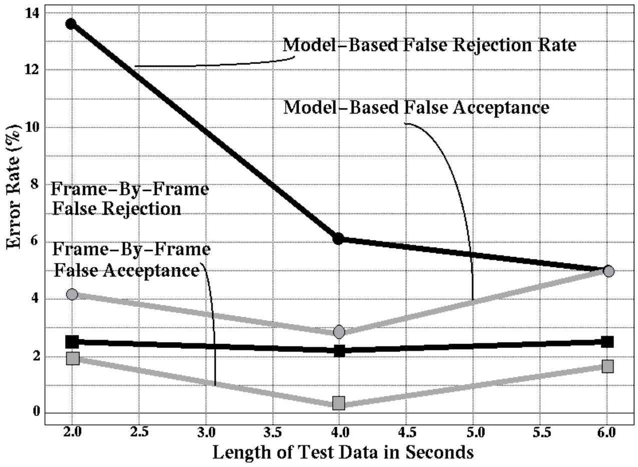

Figure 2: Speaker Verification using Noisy Data

6. CONCLUSION

Due to the parametric approach used in computing the distance between models, given by the model-based approach, to obtain rich models, sufficient amount of data should be supplied in building the models. This is another reason for making the technique suitable for doing the training and classification and not so suitable for the recognition part. As shown by the results of the previous section, increasing the amount of data used in computing the models for the model-based approach increases the accuracy of Speaker Recognition significantly. This does not affect the frame-by-frame approach as much. In computing the model for the test speaker, to obtain a good estimate of the model parameters, sufficient statistics should be present, however, comparing the frames one-by-one with the trained models, does not require as much data. For applications where enough data is present for testing the speaker (in the order of 5 seconds), the model-based test may produce better results. However, in cases where less than 3 seconds of test data is present, the frame-by-frame approach will give far better results. The computation speed of the model-based distance is significantly higher that the frame-by-frame technique. However, most of the computation of the frame-by-frame case may be done as the test data is being produced, where this is not possible with the model-based approach. In training, since more data is generally available, the model-based approach is preferred both for its accuracy and speed of comparisons. Also, in the presence of noise, even more data is necessary for improving the performance of the Model-Based system. This is probably due to the fact that some of the Gaussian prototypes will be used to model the noise. Increasing the number of Gaussians and hence increasing the amount of data is essential.

Looking at Figures 1 and 2, it is not hard to notice that the frame-by-frame approach, generally, has a lower false acceptance rate. The two approaches may be combined in the future to study the collective effect they will have on improving the system performance.

In addition, as it was presented in [5], Speech biometrics combines text-independent speaker recognition presented in this paper, speech recognition and dialog management with natural language understanding in order to simultaneously verify the acoustic characteristics of an utterance and the knowledge-based information provided in the answer to the questions. This allows for a practical Speaker Recognition system.

7. REFERENCES

[1] The Speaker Recognition API (SRAPI) Committee, “URL=http://www.srapi.com/svapi.”

[2] Homayoon S. M. Beigi and Stéphane H. Maes, and Jeffrey Sorensen, “A Distance Measure Between Collections of Distributions and Its Application to Speaker Recognition”, *Proc. ICASSP98*, Seattle, Washington, May 12-15, 1998.

[3] Homayoon S. M. Beigi and Stéphane H. Maes, “Speaker, Channel and Environment Change Detection”, *World Automation Congress (WAC), ISSCI98*, Anchorage, Alaska, May 18-22, 1998.

[4] Y. Gao, M. Padmanabhan and M. Picheny, “Speaker Adaptation Based on Pre-Clustering Training Speakers”, *EuroSpeech*, Rhodes, Greece, Sep. 22-25, 1997, Vol. 4, pp. 2095-2098.

[5] Stéphane H. Maes and Homayoon S. M. Beigi, “Open SESAME! Speech, Password or Key to Secure Your Door?”, *Asian Conference on Computer Vision*, Hong Kong, Jan. 8-11, 1998.