MULTI-ENVIRONMENT SPEAKER VERIFICATION

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

Human Language Technologies Group
IBM Research, T.J. Watson Center
P.O. Box 218, Yorktown Heights, NY 10598
e-mail address: uvc@watson.ibm.com

ABSTRACT

Here we investigate an instance of the abstract problem of pattern recognition under mismatch conditions: Models of phenomena are built with data collected in the training environment but must be used to recognize the same phenomena in another environment. The specific problem is speaker verification, where the training and testing data for each speaker can come from one of many different microphones. We use data, unlabeled with respect to channel or environment, to build, unsupervised, an easily extensible, hierarchical structure that at the finest level consists of individual speaker models, but at the coarsest level is a collection of all of the models. We then have the ability to automatically generate evolving background models from any layer of our hierarchical model when we wish to perform a verification. We give results to show that the richer our hierarchical structure, the better we do in terms of verification.

1. INTRODUCTION

We consider the problem of speaker verification under mismatched conditions when the number of environments in which training and testing data can be collected is large and variable. We describe our technique for dealing with multiple environments, making a special note of the fact that it is unsupervised and incremental, and so can potentially be used in situations where we do not know beforehand, the nature of the environments in which we will be collecting data. For example, take situations in which the only way we know about new environments is through enrollment data. In the multi-environment verification context, the technique is best described as one allowing easy modification of the normalizing background to reflect data from perhaps new and unknown environments, as those environments are seen in enrollment. Specifically, we analyze performance on data collected form 8 different microphones in a relatively noisy environment, a cafeteria. We show that we are able to obtain an improvement in multi-environment speaker recognition performance by adding information about multiple environments solely through our enrollment process, which is efficient.

Traditional approaches to such normalization in speaker verification have involved the supervised use of data from each of the observable environments to characterize them for later use in enrollment and/or testing. Rather than taking an approach which requires supervised training, and hence a prior knowledge of the environments that might be seen, we investigate an unsupervised scheme in which adaptation to data from new environments would be possible by incorporating it into our verification scheme whenever it is seen. The problem we consider is an instance of an abstract problem, namely that of pattern matching under mismatched conditions: how can we either identify a person or accept two patterns as being similar when the comparisons are or may be done under mismatched conditions (e.g. different lighting conditions or shadows for face recognition, different background scenes for object or shape recognition, different noise conditions for image recognition, different foreground and lighting noise for background texture recognition, and different reception channels for speaker recognition. As such, we see the technique itself as being useful for a number of different applications. We start with a description of the technique in the following section.

2. SOURCE VERIFICATION IN MISMATCHED ENVIRONMENTS

As a specific instance of the abstract problem, assume N sources each of who's output is received over any of M channels at any given time. For example, consider that each source is a male or female speaker and that M different types of microphones (or telephones) are the channels over which we receive their speech, also referred to here as data. Given a reception (test data) at some point in time (e.g. data from a current phone call), along with a source identity claim (e.g. the speaker's name), the task is to verify that the received data was produced by the source with the claimed identity. Since any source can be received over multiple channels (environments), any modifications that they cause in the source data must be accounted for, a procedure called environment normalization. In general the number of sources N and the number of channels M will vary as time progresses. The sources (speakers) that the system is capable of verifying comprise the enrolled target population, which is a subset of the N sources.

3. SUPERVISED ENVIRONMENT NORMALIZATION

Current approaches to channel (environment) normalization involve, in one form or another, a supervised training phase to separate and group the training and/or testing data (previous receptions from all sources) according to a predetermined set of "models" corresponding to each of the M channels. Channel dependent background models and statistics are then derived from these groups. A number of techniques exist to compare received data to the claimed source model in light of the various background models. Another approach involves trying to make the data received over any of the M channels look as if it was received over some canonical channel, thus mitigating the influence of the channel. Here again, the channels must be known so that they can be inverted. The important point is that these are techniques which require supervised training and are, in some application situations, unrealistic because of the requirement that each channel that MAY be used must be modeled and thus known ahead of time. We propose the following unsupervised approach.

4. UNSUPERVISED ENVIRONMENT NORMALIZATION

Assume that the target population, a subset of the N sources, has T elements. Our training data, used for enrolling the targets, consists of data from each of these target sources received over one of the M channels. Also, we use the data received over any of the M channels from the N-T non-target sources, as well as the targets, to model our background populations. However, rather than trying to identify the channel over which any of the sources were received, we use the following unsupervised technique in which the goal is to use data which we see only during enrollment to generate background models on the fly. In this approach any enrolled speaker can be verified, because there is no held out background population during training.

First we construct a model for each source based on its data. For generality, we let the model structure and subsequent distance measures be abstract, as this does not influence the method of selecting a background data set. Consider a partition of the original set of N sources, each of which is received over one of the M channels. At this point either a top down or a bottom up approach can be taken. The goal is to end up with an hierarchical clustering of the sources (speakers). This structure can be represented as a tree with the following property: the similarity of the leaves is proportional to the number of common ancestor nodes. In the top down method, the initial partition is one set consisting of all of the sources. Then we construct a sequence of refinements with the final one consisting of each of the singleton sources in its own subset. (refinement: partition P2 is a refinement of P1 if every element of P2 is an element of a partition of an element of P1) To construct a refinement, we need a splitting criterion which separates the sources. As mentioned before, we let this be abstract. On the other hand, this last (singleton) partition is the initial partition for the bottom up approach, where the sequence of partitions is constructed so that a partition at any point in the sequence is always a refinement of a later partition. The last partition is the initial set in the top down approach. Here, we need a grouping criterion. In either case the sequence of partitions can be represented as a tree. Assume that the tree has D levels with the root being the 0^{th} and the D^{th} consisting only of leaves. (NOTE: lower levels may have leaves as well, but they will also have nodes.) Note that the number of channels is not a parameter here, so that as we get more sources over more channels, all we need to do is grow or regenerate the tree with these additional elements.

We define a d-level cohort for any leaf l as the set of leaves with a common ancestor d levels up from the bottom (at level $D-d$) and containing the leaf l. Thus this notion of cohort is linked to the hierarchy. As more speakers are enrolled and the tree is grown or modified, the character of the cohorts automatically changes to reflect the modifications. This is because we do not pre-determine the cohorts. The benefits of this technique are apparent when we consider our implementation, described in [1], and note that we can regenerate or grow our hierarchical structure efficiently, implying efficient adaptation. In our implementation we used the bottom up approach, with a criterion which sought to match both channel and source properties. The resulting cohorts contain data which is similar due to a combination of source and channel differences. The cohort can be thus viewed as an evolving background model generator which itself was generated in an unsupervised manner.

For verification, we choose a level d_0 for the cohorts. Then given test data from a source with claimed identity i, verification is based on comparing the test data to the model for source i, which is one of the leaves of the tree, and the background model, which is derived from the cohort of the leaf corresponding to source i. If the data matches the target model better, the identity is verified, otherwise it is rejected. This comparison can be implemented as a function that takes as arguments the test data and a model, and returns a value which is an element of an ordered set. The value for the target model is compared to the value for the background model.

5. VERIFICATION DECISION FUNCTION

Denote the set of speakers by

$$\mathcal{M}_i = \{\vec{\mu}_{i,j}, \Sigma_{i,j}, p_{i,j}\}_{j=1,\ldots,n_i} = \{\Theta_{i,j}\}_{j=1,\ldots,n_i},$$

consisting of the mean vector, covariance matrix, and mixture weight for each of the n_i components of the i^{th} Gaussian Mixture Model (GMM). We use $n_i = 32$ Gaussians, obtained using the LBG algorithm, to model the training data for each speaker. The base data is 12 dimensional cepstra. The only further processing that we do is to normalize for the mean and include delta and delta-delta parameters (d is the size of the final vector). It is important to note the we do not do any form of silence or noise removal, as one of our goals is to include channel effects in our hierarchical model. We next do a bottom up binary clustering of the data based on a distance measure between models $D(\mathcal{M}_i, \mathcal{M}_j)$ described in [2].

The test data is denoted as $O = \{\vec{f}_n\}_{n=1,\ldots,N}$, and we assume that it is i.i.d. Further, we assume that the covariance matrices $\{\Sigma_{i,j}\}$ are diagonal, and write $\Sigma_{i,j}(k)$ for the variance of the k^{th} dimension. The mixture weights constitute a probability mass function on the mean vectors of any given model. Let $p_i(\vec{f_n})$ be the probability of observing frame $\vec{f_n}$ with respect to $\mathcal{M}_i$.

Given the observed testing data and an identity claim i, verification proceeds by comparing

$$\begin{aligned} \log P(O|\mathcal{M}_i) &= \sum_{n=1}^{N} \log p_i(\vec{f_n}) & (1) \\ &= \sum_{n=1}^{N} \log \left[\sum_{j=1}^{n_i} p_{i,j} p(\vec{f_n}|\Theta_{i,j}) \right] & (2) \end{aligned}$$

where, when using a Normal pdf,

$$p(\vec{f_n}|\Theta_{i,j}) = \frac{1}{(2\pi)^{d/2}|\Sigma_{i,j}|^{1/2}} e^{-\frac{1}{2}(\vec{f_n}-\vec{\mu}_{i,j})^t \Sigma_{i,j}^{-1}(\vec{f_n}-\vec{\mu}_{i,j})} \qquad (3)$$

to

$$\log P(O|\text{cohort of } \mathcal{M}_i - \mathcal{M}_i).$$

However in the experiments, we used

$$\sum_{j \,\in\, \text{cohort - i}} w_j \log P(O|\mathcal{M}_j),$$

where we chose w_j to be uniform. The verification score used in obtaining the ROC curves presented later is given by the difference of these two values. The procedure is thus text-independent.

6. TRAINING AND TESTING DATA

The 8 microphones on which we have collected training and testing data are:

- 1 = SENNHEISER
- 2 = AUDIO TECHNICA
- 3 = PRO-7A (tie clip)
- 4 = TELEX-STICK
- 5 = LAB TEC - (tie clip)
- 6 = (UNKNOWN)
- 7 = LUCENT (monitor mounted)
- 8 = RADIO SHACK (hand held)

All training data for a given speaker, i.e. that used during enrollment to create finest grain models, was collected from only one of the above 8 microphones. The testing data for that speaker was collected on the training microphone (the matched case) as well as on one of the other 8 microphones (the mismatched case). The imposter trials can be from any of the 8 microphones.

In the experiments both male and female speakers were used, however for any given piece of training or testing data, the gender was unknown. In addition, we tried to get an even distribution of microphones for training and testing. To make the experiments realistic, we limited the amount of training and testing data to approximately 10 seconds. Specifically, the average amount of training data for each enrolled model was 11.16 seconds. There were a total of 222 speakers enrolled in the final tree that was built. For the target population, we took a 28 speaker subset of the full training population (NOTE: any of the 222 could have been chosen, because we can generate backgrounds for all of them). The average amount of testing data was 9.6 seconds. There were 199 matched verification tests, 214 mismatched tests, and 382 imposter tests. The imposters were taken from a population that excluded any of the enrolled speakers.

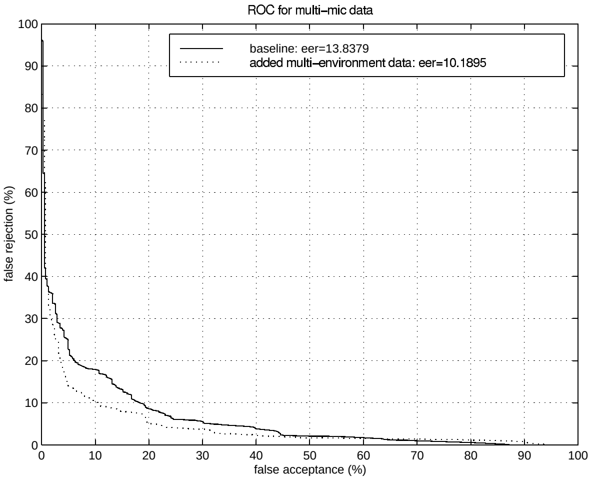

Figure 1: Effect of change in cohort character.

microphone	1	2	3	4
% of data	15.2	12.8	11.2	12.8

microphone	5	6	7	8
% of data	13.6	10.4	10.4	13.6

Figure 2: Experiment 1: 125 speaker tree

7. RESULTS

We report the results for two experiments. The effect we wanted to characterize was the change in verification performance which resulted from the change in cohort character when we added more enrollment data to the tree.

Thus the first tree that we built had 125 speakers in it, and as mentioned previously, the final tree had 222 speakers. The tables in the figures give the percentage of the enrolled speakers that represent data from each of the microphones in the experiments.

The solid curve in figure 1 gives the performance for this case. Then, we added speakers to the tree from the 8 different environments, again trying to keep the balance of the microphones the same.

The dotted curve in figure 1 gives the performance for this case. While we knew the microphone composition of the data that was enrolled in our hierarchical structure, we did not in any way use this information. The procedure to

microphone	1	2	3	4
% of data	13.96	13.52	12.61	11.26
microphone	5	6	7	8
% of data	13.96	10.81	12.16	11.72

Figure 3: Experiment 1: 222 speaker tree

modify the tree was thus unsupervised with respect to the microphone label. We notice a significant difference in the performance uniformly over the curves.

8. CONCLUSION

The basic idea that we have presented is to build an hierarchical structure purely out of speaker enrollment data and without specific knowledge of the microphone over which the data was collected. The main point is that it allows us to build background models for speaker verification on the fly whose nature changes as we get more enrollment data.

It is important to note that the sizes of the cohorts are not changing when we add, or enroll, speakers. But their character, or more precisely their composition, is changing to reflect the additional, unlabeled data. The results we have obtained indicate that we are able to exploit our efficient enrollment procedure to handle verification in multiple training and testing environments without having to resort to expensive supervised techniques.

We have also used the same technique on telephony data (results to appear elsewhere) and have observed the same behavior. We conclude, taking note that the composition of the cohort is a critical part of our verification technique, that we obtain performance gains by increasing the richness of the cohort, without having to increase its size. Further, this augmentation can be done in an unsupervised manner as a natural part of the enrollment procedure.

9. REFERENCES

[1] Homayoon S. M. Beigi, Stéphane H. Maes, Jeffrey S. Sorensen, and Upendra V. Chaudhari, "A Hierarchical Approach to Large-Scale Speaker Recognition", submitted to ICASSP'99, Phoenix, Arizona, March 15-19, 1999.

[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] T.F. Quatieri, D.A. Reynolds, and G.C. O'Leary, "Magnitude-Only Estimation of Handset Nonlinearity with Application to Speaker Recognition", Proc. ICASSP98, Seattle, Washington, May 12-15, 1998.

[4] Douglas A. Reynolds, "Comparison of Background Normalization Methods for Text-Independent Speaker Verification", EuroSpeech, Rhodes, Greece, September, 1997.

[5] Homayoon S. M. Beigi, Stéphane H. Maes, Upendra V. Chaudhari and Jeffrey S. Sorensen, "IBM Frame-by-Frame Speaker Recognition Technology", Speaker Recognition and its Commercial and Forensic Applications, Avignon, France, Apr. 20-23, 1998.