

Training Data Clustering For Improved Speech Recognition

Ananth Sankar, Françoise Beaufays, and Vassilios Digalakis
 SRI International
 333 Ravenswood Avenue
 Menlo Park, CA 94025

Abstract

We present an approach to cluster the training data for automatic speech recognition (ASR). A relative-entropy based distance metric between training data clusters is defined. This metric is used to hierarchically cluster the training data. The metric can also be used to select the closest training data clusters given a small amount of data from the test speaker. The selected clusters are then used to estimate a set of hidden Markov models (HMMs) for recognizing the speech from the test speaker. We present preliminary experimental results of the clustering algorithm and its application to ASR.

1 Introduction

While progress in ASR has been encouraging, it has become increasingly clear that ASR systems must perform well in the presence of mismatches between the training and testing environments. ASR systems trained in one environment often perform poorly in a new environment due to mismatches between the training and testing conditions. Common sources of mismatches include different transducers and channels, different speaking styles, and the presence of varying ambient and channel noise.

Even in traditional speaker-independent recognition systems that are trained and tested under similar environments, there is still a mismatch in that each test speaker is different from the training speaker population. With a sufficient amount of training data from the test speaker, we could train a speaker-dependent model that would give us the best recognition performance. Typically, however, we do not have access to such large amounts of data from the test speaker, and hence rely on using well-estimated models from a large amount of training data from similar speakers, for example, speakers from the same geographical region.

In this paper, we present a novel method to improve ASR performance under mismatched conditions. The method is based on clustering the training data to *efficiently represent the acoustic space of the training environment*, and estimating a separate set of HMMs for each cluster. During testing, the training cluster

C_i that is “closest,” in some sense, to the adaptation data is chosen, and its corresponding set of HMMs Λ_i is used to estimate the models for recognizing the test utterance. In some mismatch situations, such as speaker mismatch, this idea is appealing because the test speaker is more likely to resemble some cluster of the training speakers than the entire training set taken as a whole.

Current approaches to the problem of mismatch between the training and testing environments include adaptation methods based on maximum-likelihood (ML) estimation of the parameters of a hypothesized transformation from the training to the testing environment [1, 2], and maximum a posteriori (MAP) estimation techniques [3]. The MAP approaches perform well with large amounts of adaptation data but poorly with limited adaptation data, since, in this case, they are biased in favor of the training data. The ML-based transformation methods may be used with small amounts of adaptation data. However, their efficacy rests largely on the correctness of the hypothesized transformation, which, in most cases, is unknown.

As opposed to the MAP approach of [3], the algorithm presented in this paper uses a relatively small amount of adaptation data while efficiently organizing the training data. No assumption is made about the underlying transformation between the training and testing data, as in the ML approaches of [1, 2]. Thus, the new method is a candidate for handling a wide range of possible mismatches.

2 Training Data Clustering

The concept of clustering the training data is already used in a trivial sense in many ASR systems that use gender-dependent models. However, our approach extends this idea further by training many *template* model sets, one for each cluster of the training data.

We form the clusters of training data in the form of a hierarchical cluster tree. The leaves of the tree represent the N individual speakers in the training data, and the root represents all the training speakers. The cluster tree is built using an agglomerative clustering scheme [4], based on a relative-entropy distance metric [5]. The models we use to compute the

distance metric are mixtures of gaussian densities. A similar distance metric for speaker clustering was considered in [6]. However, in that work, the observations were assumed to be discrete, instead of continuous as in this work. Furthermore, in [6], the models used to compute the relative entropy are discrete density HMMs, whereas in our work we use mixture gaussian models.

Specifically, we first estimate the parameters of the mixture gaussian density models for each of the N training speakers. Then we compute the inter-speaker distances $D_s(m, n)$ for $m, n = 1, \dots, N$ as

$$D_s(m, n) = \frac{1}{2} [D_{\lambda_m, \lambda_n} + D_{\lambda_n, \lambda_m}], \quad (1)$$

with

$$D_{\lambda_m, \lambda_n} = \log p(\mathbf{x}_m | \lambda_m) - \log p(\mathbf{x}_m | \lambda_n), \quad (2)$$

$$D_{\lambda_n, \lambda_m} = \log p(\mathbf{x}_n | \lambda_n) - \log p(\mathbf{x}_n | \lambda_m), \quad (3)$$

where $\mathbf{x}_m, \mathbf{x}_n$ are the sets of feature vectors corresponding to speakers m and n , and λ_m, λ_n represent the mixture gaussian density models for speakers m and n .

Having computed all the inter-speaker distances, we build the tree by merging, at each iteration, the two clusters that are the closest according to some inter-cluster distance metric. Three such distance metrics [4] were considered:

$$d_{min}(C_i, C_j) = \min_{\lambda_m \in C_i, \lambda_n \in C_j} D_s(m, n), \quad (4)$$

$$d_{max}(C_i, C_j) = \max_{\lambda_m \in C_i, \lambda_n \in C_j} D_s(m, n), \quad (5)$$

$$d_{avg}(C_i, C_j) = \text{avg}_{\lambda_m \in C_i, \lambda_n \in C_j} D_s(m, n). \quad (6)$$

3 Recognition Using Cluster Tree

Once the cluster tree is built, we can train a separate HMM recognition model for each cluster. During recognition, a small amount of data from the test speaker is used to select the best cluster. Then the HMM corresponding to this cluster is used for recognition. Alternately, we can augment the test data with the data from the closest cluster in a manner similar to [7] and estimate a new model for recognition.

Estimating a separate HMM model set for each cluster node in the tree would require storage of a large number of parameters. This may well become infeasible. To address this issue, we use the maximum-likelihood transformation-based adaptation algorithm of [2] to transform the speaker-independent (SI) HMMs to the HMMs corresponding to the cluster. In this method, each set of template models Λ_i , corresponding to the training cluster C_i , is assumed to be related to the SI models, Λ_{SI} ,

through a set of parameterized transformations $T(\nu_i)$, where ν_i are the set of parameters. These parameters are estimated so as to maximize the likelihood of the training utterances X_i in the cluster C_i , given the corresponding transcriptions W_i , i.e.,

$$\nu_i = \underset{\nu_i}{\operatorname{argmax}} p(X_i | W_i, \Lambda_{SI}, \nu_i). \quad (7)$$

Thus, for each template model set, Λ_i , only the corresponding transformation parameters ν_i need to be stored, in addition to the SI model parameters. In [2], a separate affine transformation is used for each cluster of Gaussian densities in the SI models. The number of such transformations for each set of template models Λ_i can be tuned to match the available data in the corresponding training cluster.

An alternate approach to using the cluster tree is to only consider the clusters corresponding to the leaves, i.e., the individual training speakers. During testing, a small amount of adaptation data X , and the mixture gaussian models corresponding to the leaf nodes are used to compute the M most likely training speakers. These M speakers are treated as the cluster closest to the test speaker. In our experiments, we used HMM models to compute the likelihoods of the data. To avoid a separate recognition run with each of the M HMM model sets, we approximated the likelihood computation by

$$\Lambda = \underset{\Lambda_i}{\operatorname{argmax}} p(X | \Lambda_i, W_X), \quad (8)$$

where W_X is the word string decoded by a model Λ_X adapted to the adaptation data provided by the test speaker. Note that this adaptation data could be just the test sentence. Equation 8 can be interpreted as finding Λ that minimizes $D_{\Lambda_X, \Lambda}$ defined in Equation 2.

4 Experimental Results

4.1 Cluster Tree

In this section, we present an experimental study of the clustering algorithm. We experimented on the Wall Street Journal (WSJ) continuous speech recognition corpus [8]. The training set consists 142 male and 142 female training speakers.

Mixture gaussian models with 32 gaussians were trained for each of the 284 training speakers. We then used the agglomerative clustering algorithm described in Section 2 to group the training speakers. We experimented with the three distance metrics given in Equations 4, 5, and 6. We first experimented with only the 142 male speakers.

Since it is intuitive to expect well-balanced trees, we compared the three distance metrics from the point of view of how balanced the corresponding trees

were. To this effect, we used a diagrammatic representation of the trees, in which we plotted, for each tree, the number of speakers contained in each cluster vs. the cluster index (the first cluster built by the agglomerative algorithm has index 1, the second cluster has index 2, ..., the last cluster (i.e. the root node) has index $N - 1$).

The first cluster formed by the agglomerative algorithm contains two leaf nodes, and is represented in our diagram by a point of coordinates (1,2). The second cluster to be formed can either group two other leaf nodes (in which case the next point in the diagram has coordinates (2,2)), or add a leaf node to the already existing cluster (in which case the second point has coordinates (2,3)), and so on. The last point in the diagram always has coordinates $(N-1, N)$. For example, in this type of diagram, a completely unbalanced tree with $N = 8$ would be represented by the straight line:

$$(1, 2), (2, 3), (3, 4), (4, 5), (5, 6), (6, 7), (7, 8);$$

while a perfectly balanced tree would be represented by an “exponentially” growing curve:

$$(1, 2), (2, 2), (3, 2), (4, 2), (5, 4), (6, 4), (7, 8).$$

In our experiments, the d_{min} distance metric produced a very unbalanced tree, as illustrated in Figure 1(a). This can be explained by the fact that, with this distance metric, as soon as an initial cluster is formed, it becomes more likely that at the next iteration an additional speaker will be added to the existing cluster instead of two speakers outside of the cluster being merged together to form a new cluster: each speaker in the initial cluster can “attract” a close external speaker while a speaker outside the cluster can only “attract” one of its few neighbors. In contrast, the d_{max} and d_{avg} distance metrics resulted in well-balanced trees (Figures 1(b) and 1(c)).

The quality of the speaker clustering and cluster identification schemes was verified in a separate experiment. We used the hierarchical tree built using the d_{avg} distance metric to classify 10 sentences from each training speakers. These sentences were not among the sentences used to build the speaker models. We expected our algorithm to choose, for each sentence, the leaf node corresponding to the speaker who uttered the sentence. This was verified for 1323 out of the 1420 sentences, yielding an error rate of 6.8%. This error rate could be further decreased by increasing the number of gaussians used to model each speaker.

We also clustered the 142 males and 142 females using the agglomerative clustering algorithm, with the d_{avg} distance metric. On examining the two child nodes of the root of the cluster tree, we found that one contained 134 speakers, 131 females and 3 males. The second cluster had 150 speakers, 139 males, and

11 females. Thus, the error rate in sex-classification was 4.9%. This is an encouraging result.

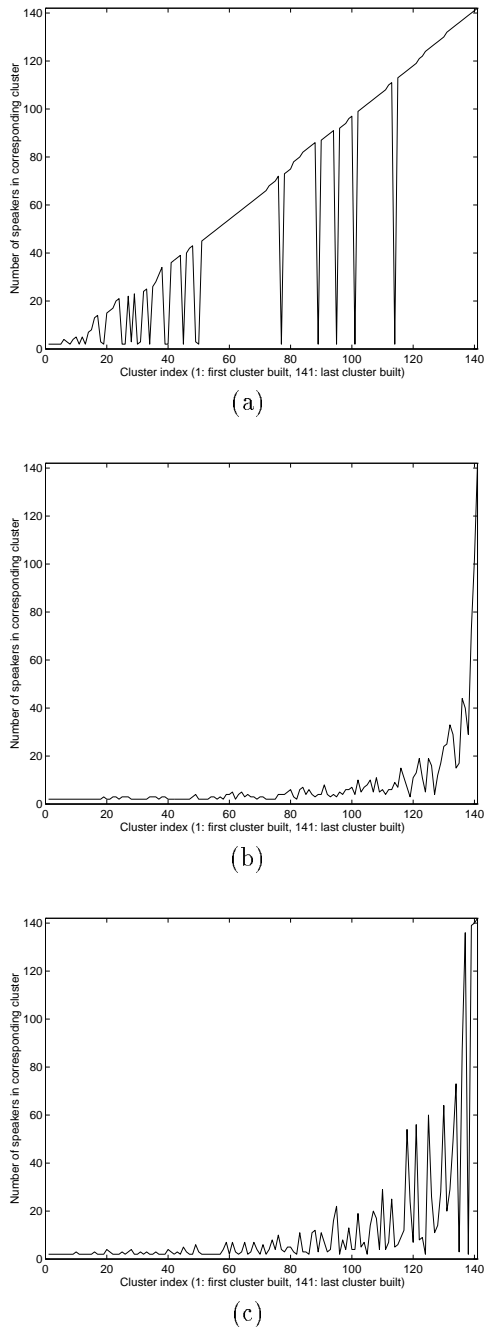


Figure 1: Clustering diagrams representing the number of speakers included in each cluster of a tree based on (a) the d_{min} distance metric, (b) the d_{max} distance metric, (c) the d_{avg} distance metric.

4.2 Recognition Using Cluster Tree

For this section we used a test set of 230 sentences from a male subset of the 1993 WSJ development and evaluation sets. We first trained SI (speaker-independent) genomic acoustic models [9] using the

18722 utterances from the 142 male speakers. In an initial experiment, we then estimated a separate adapted template model set [2] for each of the 10 test speakers. Recall that only a small number of transformation parameters need to be stored for each template model. Equation 8 was then used to identify the closest template model for each of the 230 test sentences. In all but 3 sentences, the algorithm identified the template model correctly, yielding a speaker identification error-rate of 1.3%. This shows the efficacy of using the relative-entropy measure described in Section 2.

In a second experiment, we estimated 142 adapted template models [2], one for each training speaker. For each test utterance, we computed the N closest template models using Equation 8, for several chosen values of N . We then estimated the recognition model based on these N models by averaging the means and variances of the densities of these N models. Note that this is similar to a clustering of the training speakers, except that here the clustering is done during recognition.

SI	Clustering the N template models			
	N=1	N=3	N=5	N=7
20.9	23.2	21.1	20.5	19.9

Table 1: Word Error Rate (%) Comparisons

Table 1 shows our experimental results. The first column is the SI recognition word error rate. The following columns give the recognition results for the new algorithm when estimating a recognition model by combining the N template models that are closest to the test utterance. For $N = 1$, we see that the performance is worse than that of the SI models. This can be explained by the fact that the test speaker is not well modeled by any particular training speaker, but rather by an appropriate cluster of the training speakers. This is borne out by the fact that as N increases, the word error rate decreases. For $N = 7$, the algorithm achieves about a 5% improvement over the SI models. We remark that as N increases further, the error rate will first drop, and then, as N continues to increase, the error rate will start rising, and will approach the SI error rate. This is because when $N = 142$, we are clustering all the 142 models; which is similar to the SI model. These experiments show the efficacy of selecting an appropriate cluster with the optimal value of N .

5 Summary

We have presented a novel approach based on training data clustering to improve ASR performance under mismatched conditions. Initial experimental studies have been encouraging.

We are currently studying ways to improve performance by devising better clustering techniques, and also by researching different ways of estimating the recognition models based on these clusters.

6 Acknowledgments

This work was sponsored by ARPA through Naval Command and Control Ocean Surveillance Center under contract N66001-94-C-6048.

References

- [1] A. Sankar and C.-H. Lee, "Stochastic Matching for Robust Speech Recognition," *IEEE Signal Processing Letters*, vol. 1, pp. 124–125, August 1994.
- [2] V. Digalakis, D. Rtischev, and L. Neumeyer, "Speaker adaptation using constrained reestimation of gaussian mixtures," *IEEE Transactions on Speech and Audio Processing*, 1994. to appear.
- [3] C.-H. Lee and J.-L. Gauvain, "Speaker Adaptation Based on MAP Estimation of HMM Parameters," in *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, pp. II-558–II-561, 1993.
- [4] R. Duda and P. Hart, *Pattern Classification and Scene Analysis*. John Wiley & Sons, 1973.
- [5] B.-H. Juang and L. R. Rabiner, "A Probabilistic Distance Measure for Hidden Markov Models," *AT&T Technical Journal*, vol. 64, pp. 391–408, February 1984.
- [6] J. T. Foote and H. Silverman, "A Model Distance Measure For Talker Clustering And Identification," in *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 1, pp. 317–320, 1994.
- [7] J. R. Bellegarda, P. V. de Souza, D. Nahamoo, M. Padmanabhan, M. Picheny, and L. Bahl, "Experiments Using Data Augmentation For Speaker Adaptation," in *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 1, pp. 692–695, 1995.
- [8] G. Doddington, "CSR Corpus Development," in *Proc. DARPA SLS Workshop*, pp. 363–366, 1992.
- [9] V. Digalakis, P. Monaco, and H. Murveit, "Genones: Generalized mixture tying in continuous hidden markov model-based speech recognizers," 1994. submitted to *IEEE Transactions on Speech and Audio Processing*.