Description of Acoustic Variations by Hidden Markov Models with Tree Structure

Satoru Hayamizu [a], Kai-Fu Lee
Hsiao-Wuen Hon

March 1990
CMU-CS-90-116

[a]Visiting Scholar from Electrotechnical Laboratory, Tsukuba Science City, 305 Japan.

School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

This paper provides a description of the acoustic variations of speech and its application to a speech recognition system using hidden Markov models. There are many sources of variabilities that affect the realization of a phoneme: phonetic contexts, speakers, stress, speaking rates and so on. Explicit modeling with these sources of variabilities will give more accurate and more detailed phone models, but even with a large amount of speech data, it is necessary to put some structure to the description for robustness. Tree-based HMMs are discussed as one of such structures. Three case studies are presented : HMMs with large VQ codebook sizes, decision tree clustering and speaker-clustering. They are tested on speaker-independent continuous speech recognition experiments with a 1,000 word vocabulary. Trainability and generalizability are discussed based on the experimental results.

This research was sponsored in part by U S WEST and in part by the Defense Advanced Research Projects Agency (DOD), and monitored by the Space and Naval Warfare Systems Command under Contract N00039-85-C-0163, ARPA Order No. 5167. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of U S WEST, DARPA or the US government.

Keywords : HMM(Hidden Markov Model), Binary-Tree Vector Quantization, Decision Tree Clustering, CART, Speaker Clustering, Smoothing.

Contents

1 Introduction

The purpose of this paper is to study description of acoustic variations using HMMs with tree structure and to test it in speaker independent continuous speech recognition experiments.

There are many sources of variabilities in the acoustic realization of speech, such as, phonetic contexts, speakers, stress, speaking rates and so on. One way to solve these problems is to model these acoustic variations as accurately as possible. And it will provide a more accurate and more detailed description of those variations. Context-dependent phone modeling is one example of this approach. Researchers have found that it produces very good results [Schwartz 85],[Hayamizu 88],[Lee 89a],[Bahl 89a] and clustering of context-dependent phones is successfully used to find good compromise between generability and trainability.

An allophone, which is more general than a context-dependent phone, can be defined as a phone in a particular environment. Variations for context, speaker, stress, speaking rate, etc. can be modeled as an acoustic realization in the environment [Sagayama 89], [Hayamizu 89],[Lee 90]. However, the combination of all the sources of variabilities as the environments requires astronomical amount of speech database to be able to train all the allophones sufficiently. As more sources of variabilities are taken into account, more elaborate learning is needed.

There are some hierarchies from well trained but less accurate HMMs to less trained but more accurate ones. Three case studies of the hierarchies are discussed in the following sections. First, HMMs with different VQ codebook sizes have one example of such a hierarchy. Second, there is a similar hierarchy of HMMs in the context-independent and context-dependent phones and decision tree clustering will provide a tree structure for better smoothing and prediction. And third, there is another hierarchy in the speaker-independent and speaker-dependent phones. To put a tree strucutre in the hierarchy will give better learning and smoothing. In each case, a tree structure in those hierarchies seems to give us a better description of acoustic variations.

In Section 2, general framework of description of acoustic variations and smoothing of HMMs with tree structure are discussed. In the next three sections, three case studies of this framework are described. Finally, conclusions are given in Section 6.

2 Description of Acoustic Variations

2.1 General Framework

This section provides a general framework of description of acoustic variations in speech.

Acoustic variations in speech are very complicated. One way to solve these problems is to find some invariant features from acoustic characteristics. The other way, on the contrary, is to describe those acoustic variations as accurately as possible. In general, an allophone can be defined as a phone in a particular environment [Sagayama 89], [Hayamizu 89], [Lee 90a]. The

sources of variabilities which give influence on acoustic characteristics are normally phonetic contexts, speakers, stress, speaking rates, etc.

Lets's define our symbols as:

$P = \{p\}$: a set of phones,

$E = \{e\}$: a set of environments,

$X = \{x\}$: models of acoustic realization.

For the moment, we will consider two sources of variabilities, contexts and speakers to simplify the explanation. For example, a phone $/p/$ in the context of $/p_1pp_2/$ of speaker s is written as a phone in an environment of $e = (\ /p_1pp_2/, s)$.

Description of acoustic variations is to find a mapping F_p from a set of environments E to models of acoustic realization X for each phone p,

$$F_p : E \rightarrow X$$

Note that contexts and speakers are categorical variables. They take values in a finite set and do not have any natural ordering (even the number of speakers in the world is not infinite, anyway). The most general phone model is a speaker-independent and context-independent one. The most specific phone model is a speaker-dependent and context-dependent one.

If we have enough speech database to train all the speaker-dependent and context-dependent phone models, we could have a much more accurate and detailed description of acoustic variations. But it requires a very large amount of speech data to sufficiently train all the phones in different environments.

Researchers have found clustering plays an important role to solve the trainability problem of context-dependent phones and it has led to very good results [Schwartz 85], [Hayamizu 88], [Lee 89a], [Bahl 89a]. This clustering can be written as splitting a set of environments E into the sub-sets $E_1, E_2, ..., E_n$ so as to optimize some criterion. Where:

$$E = E_1 \cup E_2 \cup E_3 \cup ... \cup E_n, \quad E_i \subset E, \quad E_i \cap E_j = \phi$$

Taking E as the triphones, E_i becomes the *generalized triphones*. There is a hierarchy which descends from well trained but less accurate context-independent phones to less trained but more accurate generalized triphones. The clustering technique gives a good compromise of generability and trainability.

In view of this success with clustering technique, a natural question might be "what about putting a more general struture in description ?" We will see other types of hierarchy and tree structures in the next three sections. In common, the hierarchy is utilized in two ways.

- train rough models first and then train the detailed models starting from the rough ones.
- smooth detailed models with rough ones to get robustness.

The first one is rather easy to implement. We will discuss the second aspect in greater detail, i.e., smoothing HMMs with tree structures.

2.2 Smoothing HMMs with Tree Structure

Suppose the description of acoustic variations have a tree structure like Figure 2.1, where $E_0 = E$, $E_1 \cup E_2 = E_0$, $E_1 \cap E_2 = \phi$, $E_3 \cup E_4 = E_1$, $E_3 \cap E_4 = \phi$, $E_5 \cup E_6 = E_2$, $E_5 \cap E_6 = \phi$.

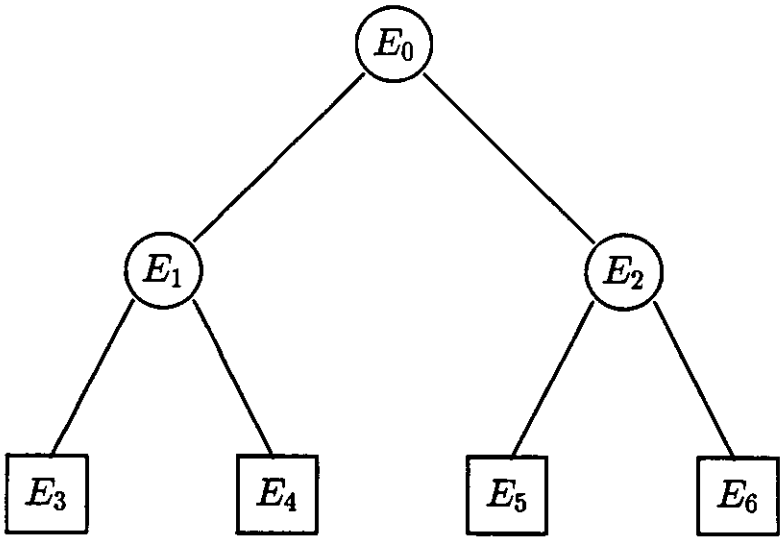

Figure 2.1. A sample tree structure for description of acoustic variations

This tree can be generated by either top-down splitting or bottom-up merging. The root node is the most well-trained but least accurate and the leaf nodes are not as well-trained but more accurate. The internal nodes are intermediates between the root node and the leaf nodes. The internal nodes are not as well-trained but more accurate than the root node. And they are better-trained but still less accurate than the leaf nodes. There might be any number of levels in the tree sturcture.

The idea here is to use all the ancestors of each node for smoothing HMM. It is expected that a better smoothing will be obtained using these internal nodes than just using only the root and leaf nodes. One node is smoothed using a linear combination of all the nodes in the path from the node smoothed to the root node. Also a special node of uniform distribution is added on the top of root node in order to avoid a zero probability. In the example of Figure 2.1, node X_3 (HMMs in the environment E_3) is smoothed using a linear combination of X_3,

X_1, X_0 and U,

$$X_3' = \lambda_1 X_3 + \lambda_2 X_1 + \lambda_3 X_0 + \lambda_4 U$$

where $\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 = 1$. Estimation of λ_i is done by *deleted interpolation* [Jelinek 80]. It is to divide the training data into several blocks, use all the blocks except a *deleted* block to estimate λ_i on that block, and average the λ results. For the experiments in this paper, we divided the training data into two blocks during the last iteration and maintained separate output and transition counts for each block. Then, deleted interpolation was run phone by phone to estimate λ_i for all the nodes. For the example of Figure 2.1, there are seven nodes and seven paths ($E_3 - E_1 - E_0 - U$, $E_4 - E_1 - E_0 - U$, $E_5 - E_2 - E_0 - U$, $E_6 - E_2 - E_0 - U$, $E_1 - E_0 - U$, $E_2 - E_0 - U$, $E_0 - U$) to be smoothed. Also, we estimated λ_i for three distributions (begin, middle and end) of each phone along all the paths independently.

2.3 Three Case Studies

Three case studies with SPHINX system will be presented;

1. HMMs with large VQ codebook sizes
2. Decision Tree Clustering
3. Speaker Clustering

A common feature in all three case studies is to use a tree structure to describe the variations and to smooth the HMMs with tree structures. Also, all of them are then tested on speaker-independent continuous speech recognition experiments.

3 HMMs with Large VQ Codebook Size

3.1 Smoothing HMMs with Different VQ-Sizes

In dicsrete HMMs, the distortion of vector quantization is highly related to the accuracy of its acoustic property. As less the distortion is, more accurate of modeling is possible. Table 3.1 shows the distortions of cepstral coefficients for codebook sizes 2 to 256.

Table 3.1 Distortion and codebook size of cepstral coefficients

codebook size	2	4	8	16	32	64	128	256
distortion	0.89	0.61	0.48	0.36	0.29	0.24	0.19	0.16

A natural question to ask is what happens when the codebook size is increased from 256 to more. The problem is that if we increase the codebook size of the HMMs, we will need more data to train them to the same level as before. HMMs with different VQ sizes will provide better smoothing for that purpose. But this means the codewords of different codebook sizes must be related. Binary tree searched vector quantization makes it possible to make codewords

in different codebooks related. This is explained later in Section 3.2. For example, HMMs of codebook size 4096, X_{4096} can be smoothed with HMMs of codebook-size 1024, X_{1024}, those of codebook size 256, X_{256} and uniform distribution U as:

$$X'_{4096} = \lambda_1 X_{4096} + \lambda_2 X_{1024} + \lambda_3 X_{256} + \lambda_4 U$$

where $\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 = 1$. In HMMs of codebook size 4,096, four codewords are tied to those for codebook size 1,024 and sixteen codewords are tied to those for codebook size 256.

If we use generalized triphones for the description of contexts, there is another hierarchy of context-independent to context-dependent phones in addition to the hierarchy of VQ codebook sizes (Figure 3.1). Then the smoothing equation becomes:

$$X'_{1,024-CD} = \lambda_1 X_{1,024-CD} + \lambda_2 X_{256-CD} + \lambda_3 X_{1,024-CI} + \lambda_4 X_{256-CI} + \lambda_5 U$$

where $\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 + \lambda_5 = 1$. We smooth all the nodes along the path independently.

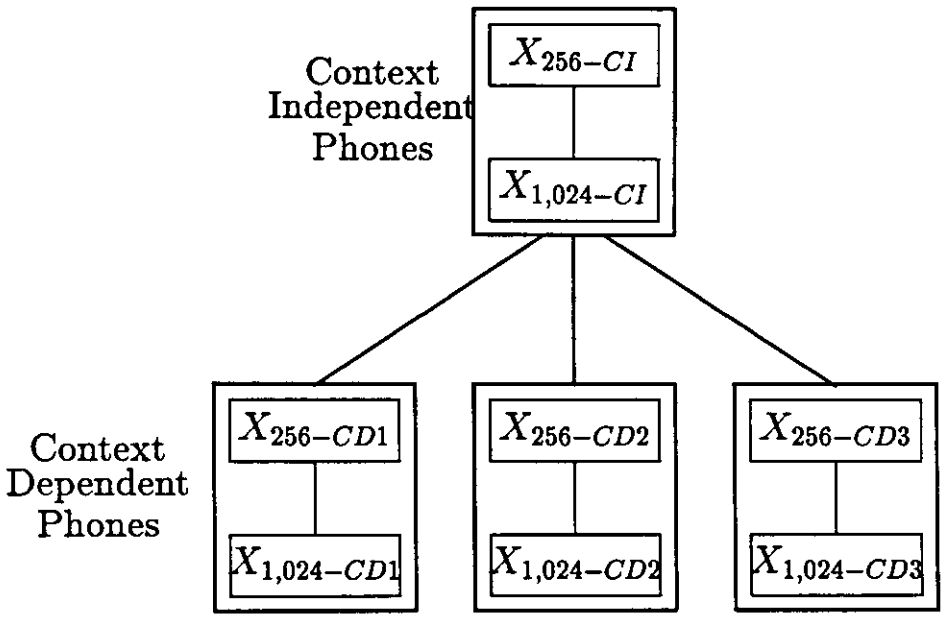

Figure 3.1. Two hierarchies of context independent-dependent and VQ codebook sizes

3.2 Binary Tree Searched Vector Quantization

Binary tree searched vector quantization [Gray 82] makes it possible to make the codewords in different codebooks related. We can therefore make a large size of VQ codebooks starting from a smaller size and then limiting them within the cluster. All the codewords in the large size of VQ codebooks can then have their correspondent codewords in a smaller size of VQ codebooks.

For example, let the cepstrum coefficients be clustered into 256 clusters by assigning them to the nearest of 256 codewords $c(i), i = 1 \sim 256$. Then each cluster is splitted into two clusters (total 512 clusters) and let centroids of 512 clusters form 512 new codewords $c'(i), i = 1 \sim 512$. If one new vector is quantized as one of these 512 codewords (say, $c'(15)$) in the codebook of 512, it could also be quantized as the correspondent codeword of 256 codewords (say, $c(8)$) in the codebook size of 256. This means that HMMs with smaller codebook size can be easily calculated from HMMs with larger codebook size if codebooks are made by this method.

Currently, vector quantization is not as costly as the training of HMMs or recognition. But designing codebooks takes long time for the large codebook size. Binary tree searched vector quantization is known to be a fast algorithm. For designing a codebook, it has a computational cost which is only linear with the number of bits for the codebook size. For example, a fully searched VQ, with a size of 4,096, is 16 times more costly than that of 256. For binary tree searched VQ, with a size of 4,096, only 1.5 times the computational time is required. That is another reason to use this algorithm here. Table 3.2 shows the distortions for codebook sizes of 512 to 4,096 generated by this algorithm.

Table 3.2 Distortion and codebook size of cepstral coefficients

codebook size	512	1,024	2,048	4,096
distortion	0.14	0.12	0.11	0.09

3.3 Experiments and Results

The database used here is the speaker-independent DARPA Resource Management database [Price 88]. The task is a 991-word continuous speech task. The word pair grammar (perplexity 60) was used with no corrective training. The test set consists of 320 sentences from 32 speakers randomly selected from the 1988 and 1989 test sets. The training set consists of 4,358 sentences from 109 speakers.

A cepstrum analysis of order 14 is done and 32 LPC cepstral coefficients are calculated by a recursive equation. Then bilinear transformation is done resulting in 12 mel-scaled cepstral coefficients. Frame shift is 10 msec and sampling frequency is 16 kHz. Three codebooks of cepstrum, delta-cepstrum, power-and-delta-power are used. The recognition system is identical to the SPHINX of [Lee 89a] unless otherwise specified.

The recognition results using the 48 context independent phones are shown in Table 3.3. Codebook sizes of 256 (standard SPHINX), 512, 1,024, 2,048, 4,096 are used. HMMs were

smoothed with smaller codebooks of HMMs. Deleted interpolation was done based on the count ranges in these experiments. Note, a codebook size of 2,048 actually gave a 1.0% improvement in word accuracy (about 5% error reduction).

Table 3.3 Recognition results for codebook size of 256, 512, 1024, 2048, 4096 (HMMs are smoothed using those with different codebook sizes).

size	percent correct(word accuracy)	smoothing
256	83.0% (80.8%)	baseline
512	83.5% (81.0%)	with 256 and uniform
1,024	83.8% (81.3%)	with 256 and uniform
2,048	84.2% (81.7%)	with 256 and uniform
4,096	84.2% (81.6%)	with 256 and uniform
2,048	84.1% (81.8%)	with 256,512,1024,uniform
4,096	84.2% (81.7%)	with 256,1024,uniform

Table 3.4 shows the results where HMMs were smoothed with a uniform distribution. As the codebook size increases, more differences become apparent between HMMs with and those without internal nodes in the smoothing.

Table 3.4 Recognition results for codebook size of 512, 1024, 2048, 4096 (smooth HMMs with uniform).

size	percent correct(word accuracy)	smoothing
512	83.9% (81.4%)	with uniform
1,024	83.8% (81.2%)	with uniform
2,048	83.3% (80.8%)	with uniform
4,096	83.4% (80.6%)	with uniform

Table 3.5 shows the recognition results using between-word modeling for allophones (total 1100 general triphones) [Lee 89b] for codebook sizes of 256 and 1024. The results of a codebook size 1024 are shown in Table 3.5. In this case, the HMMs were smoothed (1) with codebooks of 256 and uniform distribution (2) with only uniform distribution. The word accuracies are almost same for both codebook sizes and there is a 0.7% difference between word accuracies with and without internal nodes in the smoothing.

Table 3.5 Recognition results for codebook size of 256 and 1024 using 1,100 generalized triphones.

size	percent correct(word accuracy)	smoothing
256	94.2% (93.0%)	baseline
1,024	94.2% (93.1%)	with 256 and uniform
1,024	93.6% (92.4%)	with uniform

3.4 Discussion

Binary tree searched vector quantization is used to make different size of codebooks being related to each other. Codebook size of 2,048 gave a 1.0 % improvement in word accuracy (about 5% error reduction) using context independent phones. The word accuracies are almost the same for the codebook sizes of 256 and 1,024 by using 1,100 generalized triphones. For both cases, there are some differences between HMMs with and those without internal nodes in smoothing.

After all, increasing the VQ codebook size does not help very much, though smoothing with internal nodes does offer significant results. Seeing that distortions get smaller as increase of codebook sizes, there still remains some rooms of being more accurate modeling of acoustic realizations. However, we think a larger database is still necesssary in order to get enough training for larger VQ codebook sizes.

4 Decision Tree Clustering

4.1 Decision-Tree-based Context Clustering

The agglomerative clustering used in [Lee 89a] is excellent from the point of view of minimizing entropy. However, it has two drawbacks : smoothing must be done with context-independent phones alone and it cannot predict about the unknown contexts. This suggests the necessity of imposing some structure in the context-independent and context-dependent hierarchy.

Decision-tree-based context-clustering will provide a tree structure for better smoothing and prediction. The idea is to use the features of neighbouring phones to guide context clustering so that the acoustic realization of unknown contexts can be analogically predicted using these features. Also, having internal nodes will provide better smoothing.

Decision trees [Breiman 84] have been used to get statistical language models [Bahl 89b] and to cluster phones into broad classes. Here the technique is applied to context clustering [Bahl 89a], [Lee 90a]. In the agglomerative clustering, description of variations has a simple hierarchy of context-independent and generalized-triphones. In the decision tree clustering, description of variations has this more general tree structure.

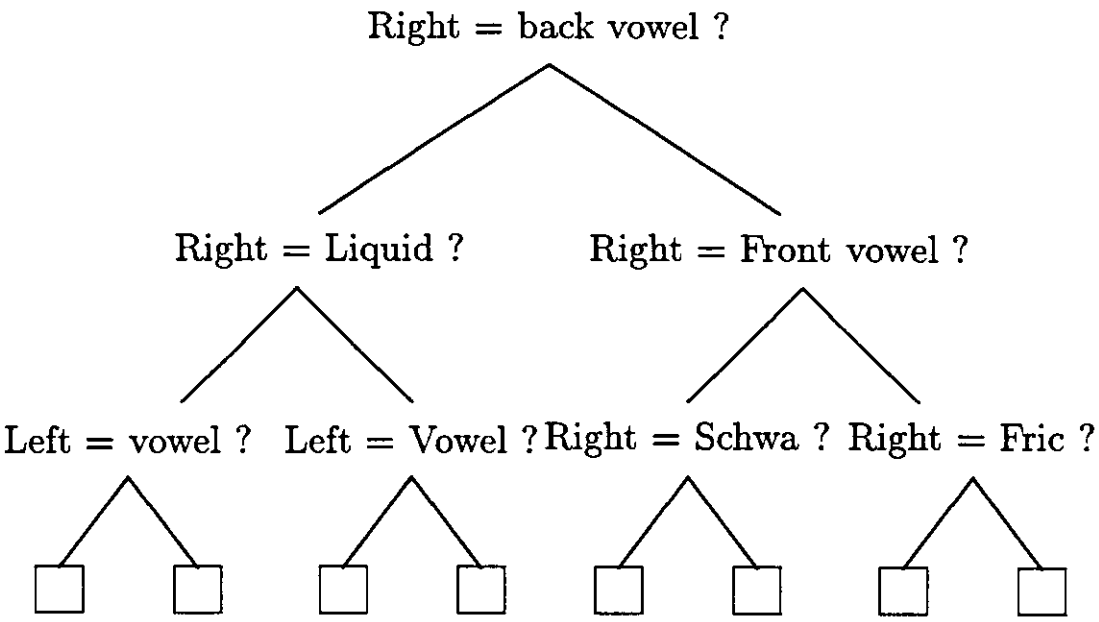

Figure 4.1. An example of a decision tree that clusters the contexts of the phone /k/

Figure 4.1 shows an example of decision tree that clusters the contexts of the phone $/k/$. Each node has a binary "question" about contexts of the allophones, for example, "is the previous phone a front vowel ?". Each node represents a sub-set of contexts according to the questions in the path from the root node to itself.

The distance metric used for this clustering is identical to that for agglomerative clustering [Lee 89a],

$$D(a,b) = P(m)H(m) - P(a)H(a) - P(b)H(b)$$

$$H(x) = -\sum P(c|x)\, logP(c|x)$$

where $D(a,b)$ is the distance between two models of a and b, $H(x)$ is the entropy of the distribution in model x, $P(x)$ is the frequency (or count) of a model, and $P(c|x)$ is the output probability of the codeword c in model x. In measuring the distance between two models, we only consider the output probabilities, and ignore the transition probabilities, which are of secondary importance. This information-theoretic distance measure has been shown to be equivalent to a maximun likelihood metric [Lee 89a]. Other details about decision tree clustering are found in [Lee 90a].

4.2 Smoothing CART-HMMs

For robustness, every node in the decision tree (CART-HMMs) needs some smoothing. All the nodes in the tree are smoothed along all the paths from the root node to the node which is to

be smoothed. Also, the special node of uniform distribution is added on the top of the root node in order to avoid a zero probability. The node smoothed is then represented by a linear combination of all the nodes along the path. For the example of Figure 4.1, smoothing is done as:

$$X_3' = \lambda_1 X_3 + \lambda_2 X_1 + \lambda_3 X_0 + \lambda_4 U$$

where $\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 = 1$. Estimation of λ_i is done by deleted interpolation. All the leaf nodes of the tree are trained and the internal nodes are calculated by summing up those child nodes. The training data is divided into two blocks during the last iteration. Separate output and transition counts are maintained for each block. Then, deleted interpolation is run phone by phone to estimate λ_i for all the nodes. We estimate λ_i for three distributions (begin, middle and end) of each phone along all the paths independently.

4.3 Experiments and Results

The task tested here is still the speaker independent 991-word vocabulary continuous speech recognition of DARPA Resource Management database. The word pair grammar (perplexity 60) was used. Acoustic analysis was the same as Section 3.3. A total of 1,800 leaf nodes (HMMs) were generated by the decision tree clustering and agglomerative clustering and used for the recognition experiments. Between-word modeling for allophones and corrective training were not used, thus the error rates are 30-50% higher than the current best version of the SPHINX system.

We tested the decision tree clustering on vocabulary independent recognitions [Hon 89], [Hon 90]. The General English training set consists of 3,000 TIMIT sentences, 2,000 Harvard sentences, and 10,000 General English sentences which are collected at Carnegie Mellon. The total 15,000 training sentences cover about 90% of the triphones in the test set.

The test set consists of a TI-test and a CMU-test. The TI-test set consists of 320 sentences from 32 speakers (a random selection from the 1988 and 1989 test sets). The CMU-test set consists of 320 sentences (same sentences as above ones) from 32 speakers (different speakers) recorded at Carnegie Mellon.

Table 4.1 Recognition results for the CMU-test set using the General English training sentences

	percent correct (word accuracy)
agglomerative clustering	90.4% (88.9%)
decistion-tree clustering	90.4% (89.2%)

Table 4.1 shows the preliminary recognition results for the CMU-test set using the General English training sentences. The error rate of decision-tree clustering is comparable for that of agglomerative clustering [Lee 89a].

The current triphone coverage of 90% may become larger if it is weighted by frequency and the missing 10% contexts may not be important. So, we tested less covered vocabulary by using the TIMIT training sentences alone. Table 4.2 shows the recognition results for the TI-test set using the TIMIT training sentences. The word accuracy of decision-tree clustering is 0.9% better than that of agglomerative clustering (5% error reduction). These results indicate decision tree clustering is powerful, particularly for vocabulary independent situations.

Table 4.2 Recognition results for the TI-test set using the TIMIT training sentences

	percent correct (word accuracy)
agglomerative clustering	84.8% (82.0%)
decistion-tree clustering	85.4% (82.9%)

4.4 Discussion

The recognition results show that decision-tree-based context-clustering works well to describe acoustic variations about contexts. We believe that it also predicts well about unknown contexts.

However, there is a gap between the error rates for vocabulary dependent and vocabulary independent training sets in the case of the test set recorded at TI [Lee 90a]. The gap is probably due to the difference in recording conditions between the training and test materials. It indicates that some kind of noise reduction or adaptation to the recording environment [Acero 90] is necessary to fill the gap between two training sentences.

5 Speaker Clustering

5.1 Top-Down Speaker Clustering

This section discusses some aspects of description of acoustic variations for the speakers. Speaker variability has been studied in the speaker recognition and speaker adaptation. In the speaker independent speech recognition, dynamic programming based system using whole word multiple templates had been studied in the 1970's. Recently there are some studies on speaker clustering in a dicrete HMM-based sytem [Lee 89a] (agglomerative clustering) and in a continuous HMM-based system [Rabiner 89]. In this section, we discuss three aspects of speaker clustering: top-down splitting, cross validation and smoothing speaker clusters with tree structures.

First we discuss speaker clustering by top-down splitting. The algorithm used here is a variant of Linde-Buzo-Gray algorithm [Linde 80] for vector quantizer design and given as follows.

1. Merge all the speakers into one cluster S_1, $m = 1$.
2. Split one speaker cluster $S_i = \{s_i(j), j = 1, n_i\}$ into two clusters ($m = m + 1$)
   (a) Find the farthest speaker $\hat{j}$ from the centroid of speaker cluster
   (b) Order all the speakers in the cluster according to the distance from $\hat{j}$. Let $\hat{j}$ be $o(1)$, nearest to $\hat{j}$ be $o(2)$ and farthest to $\hat{j}$ be $o(n_i)$, $\{o(j), j = 1, n_i\}$.
   (c) Find minimum lost information for all the splits along the order. $E_1 = \{s(o(1)), s(o(2)), ..., s(o(k))\}$ and $E_2 = \{s(o(k+1)), ..., s(o(n_i))\}$, $1 < k < n_i$.
3. Merge all the speakers in each cluster (centroid).
4. Assign all the speakers to the nearest speaker clusters.
5. Till convergence repeat 3-4.
6. If $m = M$ (number of clusters to be splitted) then stop
7. Find a speaker cluster to be splitted next which has maximum lost information and go to 2.

The distance metric of two models is the same as that used in the decision tree clustering. Here, HMMs for each speaker are trained using sentences spoken by the speaker. Only output probabilities were considered in the clustering and each HMM had three distributions (begin, middle, end) of three codebooks (ceps, δceps, power-δpower).

Figure 5.1 shows the relationship between the number of speaker clusters and the average lost information for the context independent phones of "AE" and "K" of 109 speakers in the Resource Management training sentences.

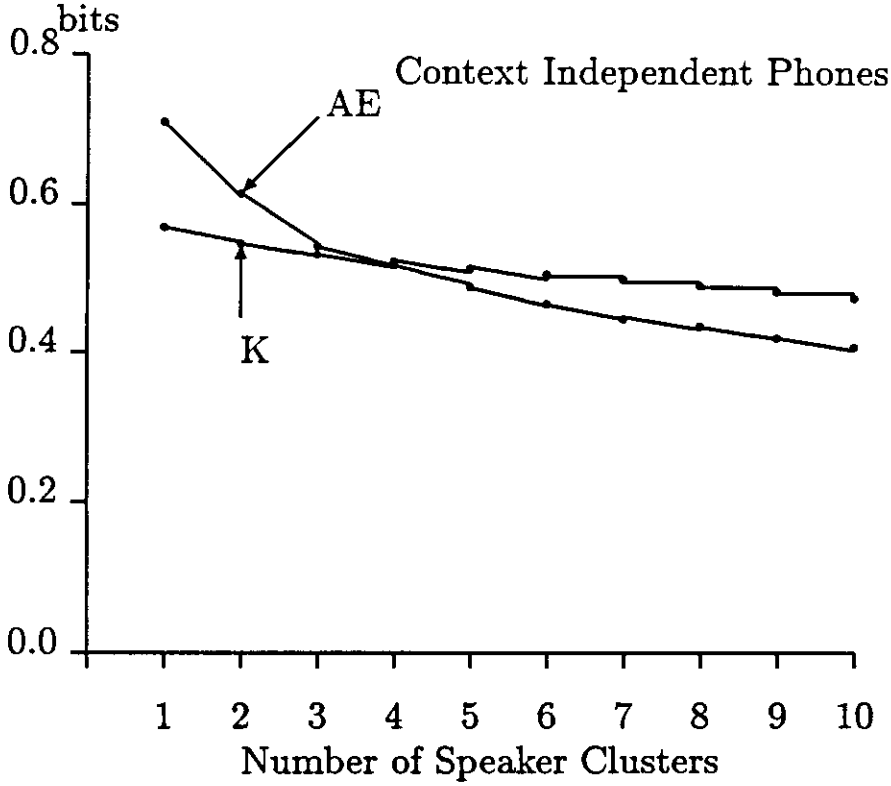

Figure 5.1. Lost information at speaker clustering of context independent phones

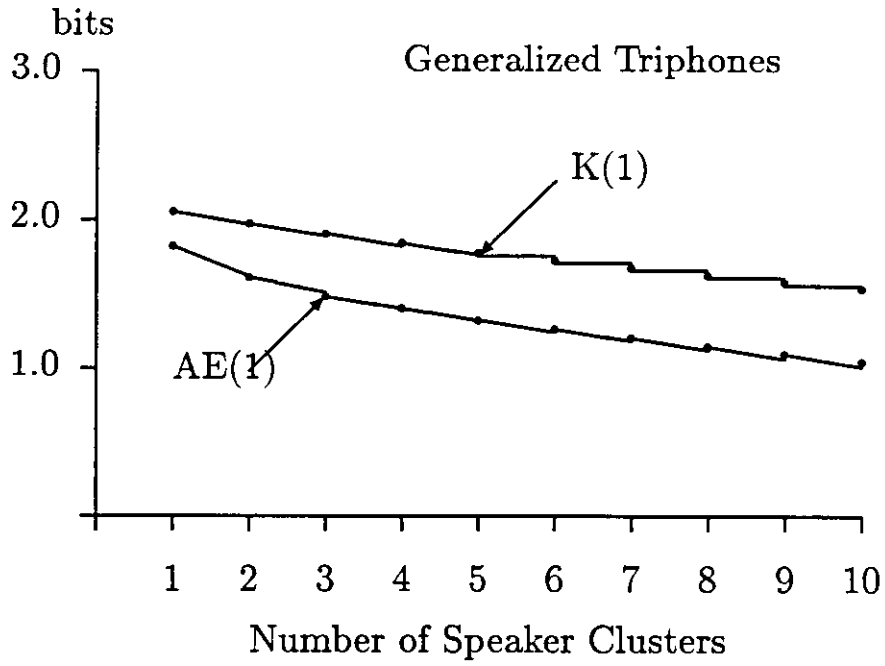

Figure 5.2. Lost information at speaker clustering of generalized triphones

Figure 5.2 shows the same results for two samples of generalized triphones "K(1)" and "AE(1)". It is observed that lost information decreases constantly as the number of speaker clusters increases. It seems we can get more detailed models as we increase the number of speaker clusters.

However, these results do not stand up if we test them by cross validation [Breiman 84]. Figure 5.3 shows the relationship between the number of speaker clusters and the probabilities that a speaker cluster produces each speaker model with and without cross validation. Probabilities for the generalized triphone of "AE(1)" out of 1,100 generalized triphones are shown. They are approximated by simply using counts and averaged for all the speakers. Without cross validation, speaker cluster contains the speaker to be tested. With cross validation, the speaker is eliminated when merging speakers into the speaker cluster. For generalized triphones, probabilities increase constantly without cross validation. But those with cross validation decrease after three speaker clusters.

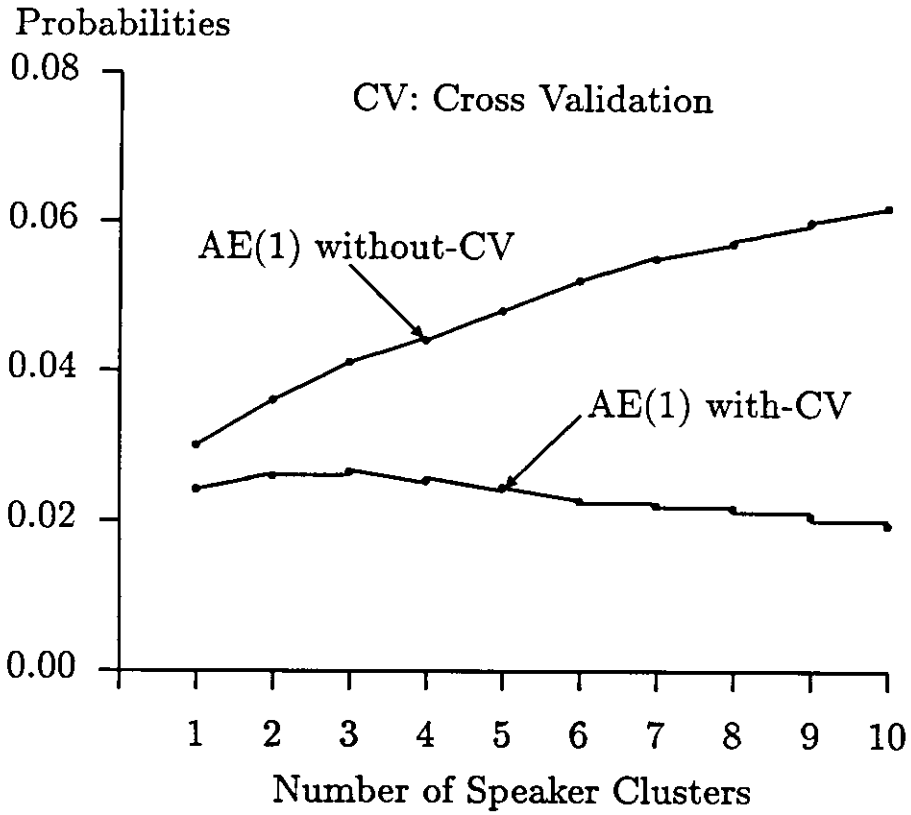

Figure 5.3. Probabilities at Speaker Clustering of Generalized Triphones [AE(1)].

These figures suggest that we can expect some improvement using two or three speaker clusters. They also suggest if we use more than four speaker clusters it may not be trained well enough. For less trained models like generalized triphones, the difference between having and not having cross validation is larger than for context independent phones. These results show the importance of cross validation in the clustering of less trained models.

5.2 Smoothing of Speaker-Cluster HMMs

With the combination of contexts and speakers as two sources of variabilites, there are two hierarchies, one is context-independent (CI) and generalized-triphone (CD) hierachy and the other is speaker-independent (SI) and speaker-cluster-dependent (SD) hierarchy. In that case, a description of the variations has the tree structure shown in Figure 5.4.

combination of all the nodes from the root node to the node to be smoothed.

$$X'_{CD,SD} = \lambda_1 X_{CD,SD} + \lambda_2 X_{CD,SI} + \lambda_3 X_{CI,SI} + \lambda_4 U$$

where SI is Speaker-Independent, SD is Speaker-Cluster-Dependent, CI is Context-Independent, CD is Context-Dependent (Generalized Triphones), and $\lambda_1 + \lambda_2 + \lambda_3 + \lambda_4 = 1$. Estimation of λ_i is done by deleted interpolation. All the leaf nodes of the tree are trained and the internal nodes are calculated by summing up those child nodes. The training data is divided into two blocks during the last iteration. Separate output and transition counts are maintained for each block. Then, deleted interpolation is run phone by phone to estimate λ_i for all the nodes. We estimate λ_i for three distributions (begin, middle and end) of each phone along all the paths independently.

5.3 Experiments and Results

To test the speaker clustering algorithm and the smoothing method above, speaker independent continuous speech recognition experiments were conducted. These consist of speaker clustering, training of each speaker cluster, smoothing and recognition.

First, we conducted a top-down clustering of speakers. We used 4,358 sentences from 109 speakers (about 40 sentences per speaker) to train HMMs for each speaker. For the speaker clustering, we used only 47 context-independent phones (silence is excluded) of 109 speakers. We splitted the 109 speakers into two and three speaker clusters. Table 5.1 shows the male/female composition of two and three clusters. Most clusters are dominated by male or female as the bottom-up speaker clustering [Lee 89a].

Table 5.1 Male/female composition of two and three speaker clusters as produced by a top-down clustering algorithm

Two Clusters		
	Male	Female
Cluster No.1	78	0
Cluster No.2	3	28

Three Clusters		
	Male	Female
Cluster No.1	42	0
Cluster No.2	36	0
Cluster No.3	3	28

Using these clusters, all speaker clusters were trained using same database of 4,358 sentences from 109 speakers. A total of 1,100 generalized triphones using between-word modeling was used to represent variations for contexts. We used speaker independent models as the initial models for the training and one iteration of forward backward training was run. We used the

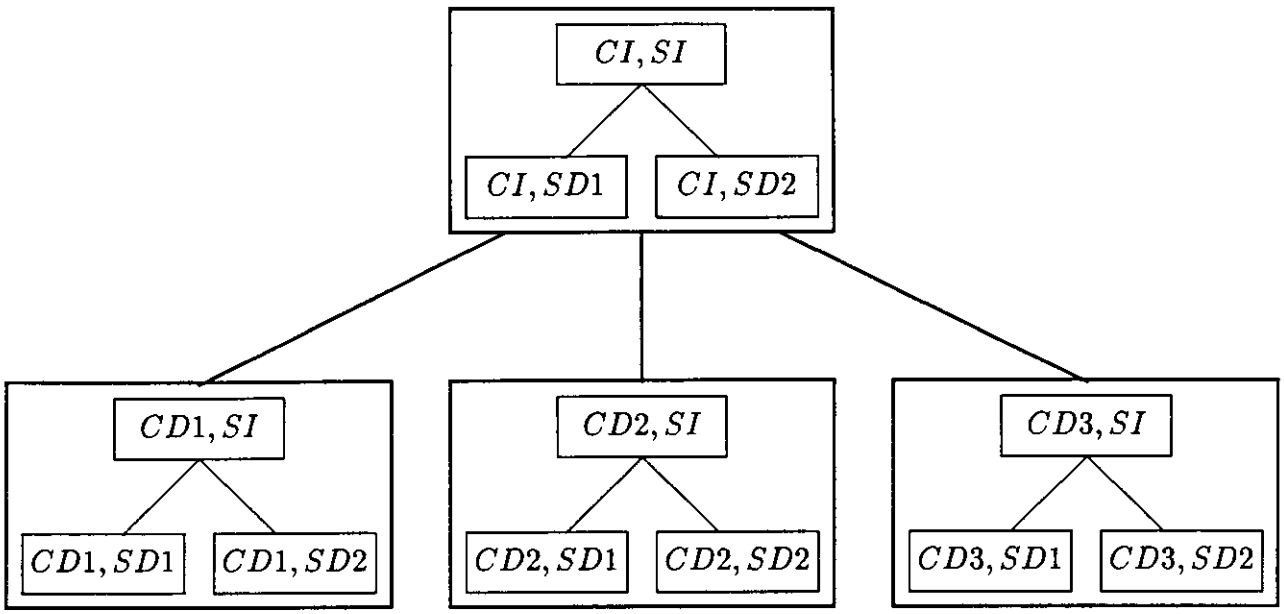

Figure 5.4 Description of acoustic variations for contexts $(CI - CD)$ and speakers $(SI - SD)$.

Acoustic variabilities for speakers are described by multiple speaker clusters. To use these multiple speaker clusters for the recognition, smoothing is necesary to get more robust models. Out of the two sources of variabilities, the variations for the contexts are supposed to be greater than those for the speakers. So, we simplify the tree of Figure 5.4 to the following tree (Figure 5.5).

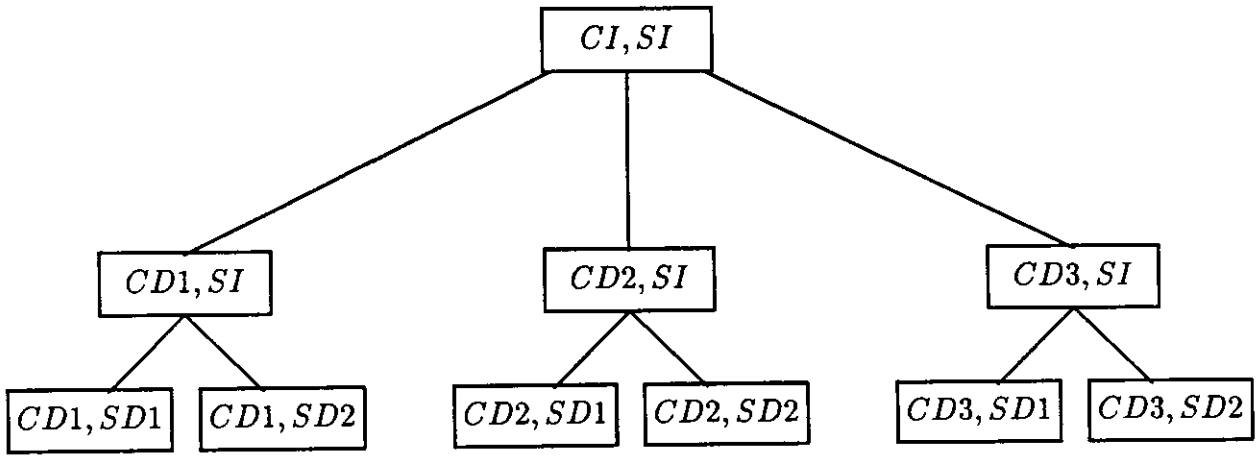

Figure 5.5 A simplified tree of description of variations for contexts $(CI - CD)$ and speakers $(SI - SD)$.

Also, a special node of uniform probabilities is added to the top of root node. Then, we can smooth all the nodes in the tree as in Section 3.3. Smoothing is conducted by the linear

same VQ codebooks for all the speaker clusters. For the deleted interpolation, each speaker cluster was divided into two sets and both were trained separately. Then, smoothing was conducted as explained in Section 5.2. We estimated λ_i for the three distributions of each phone along all the paths independ ently.

The task tested here is a speaker independent 991-word vocabulary continuous speech recognition of DARPA Resource Management database. Both cases were tested with word pair grammar (perplexity 60) and without grammar. Acoustic analysis was same as Section 3.3. Corrective training was not used. The test set consits of 320 sentences from 32 speakers randomly selected from the 1988 and 1989 test sets. To select an appropriate speaker cluster for each speaker, we simply selected a speaker cluster which produced the best recognition score from all the speaker clusters. Table 5.2 shows the recognition results using two and three speaker clusters.

Table 5.2 Recognition results using two and three speaker clusters.
Results shown are percent-correct (word-accuracy).

Word Pair Grammar	
Speaker Independent	94.2% (93.0%)
2 speaker clusters	94.7% (93.4%)
3 speaker clusters	94.6% (93.4%)
No Grammar	
Speaker Independent	75.3% (72.2%)
2 speaker clusters	77.4% (74.6%)
3 speaker clusters	76.8% (73.8%)

Using two speaker clusters, we obtained about 6% (word pair grammar) and 9% (no grammar) error reduction. These results show the potential of speaker clustering.

5.4 Discussion

Both clustering results and recognition results indicate that we can expect to have better description of acoustic variations using only two or three speaker clusters. But why is it that increasing the number of speaker clusters does not give a better description ? It may be due to several reasons. First, we need a larger database to train each speaker cluster sufficiently. In smoothing, many λ_i for the speaker-cluster-dependent phones were very small and this indicates that those context-dependent and speaker-cluster-dependent phones were not trained well enough. Secondly, we may need to use speaker cluster dependent VQ codebooks, even though HMMs with a universal VQ codebook seem to represent most of the speaker variabilities.

Huang had obtained almost the same results as two speaker cluster case by training male and female separately and by testing them for known gender [Huang 90]. This result also supports the potential of speaker clustering.

Another important issue derived from this study is the importance of cross validations. It is observed that lost information decreases constantly as the number of speaker clusters increases and it seems we can have more detailed models as we increase the number of speaker clusters. But it does not hold up if we check it by cross validation as we saw in Section 5.1.

6 Conclusion

In this paper, we have presented a description of the acoustic variations using HMMs with tree structure. The general framework of this was given and discussed in three case studies.

First, HMMs with different VQ codebook sizes were studied. Binary tree searched vector quantization was used to make different size of codebooks being related to each other. A codebook size of 2,048 gave about a 5 % error reduction for the case of context independent phones. Smoothing HMMs with different size of codebooks also gave us more robust modelings.

Second, decision tree clustering was presented to provide a tree structure for better smoothing and prediction about unknown contexts. The recognition results were comparable for the General English training set and about a 5% error reduction for the TIMIT database. Decision tree clustering is shown to be powerful, particularly for vocabulary independent situations.

Finally, speaker clustering was studied. An algorithm for top-down clustering of speakers was given. Also, the importance of cross validations for speaker clustering was shown. Using two speaker clusters with 1,100 generalized triphones, we obtained about 6% (word pair grammar) and 9% (no grammar) error reduction and these results show the potential of speaker clustering.

The point which all three cases share is that a tree structure in several hierarchies will give us a better description of acoustic variations. We believe a larger database is still necesssary to get enough training for more detailed description of acoustic variations.

Acknowledgments

The authors would like to thank Professor Raj Reddy for his encouragement and support and would like to thank Dr. Robert Weide for providing the speech database and phonological knowledge for the decision tree clustering. The authors would like to thank Mr. Cecil Huang and Mr. Jonathan Swartz for providing software for decision tree clustering. The authors would also like to thank Miss Jeanette Dravk for reading this paper.

References

[Acero 90] Acero, A., Stern, R.M., "Environmental Robustness in Automatic Speech Recognition", *IEEE International Conference on Acoustics, Speech and Signal Processing*, April

1990.

[Bahl 89a] Bahl, L.R., et. al., "Large Vocabulary Natural Language Continuous Speech Recognition", *IEEE International Conference on Acoustics, Speech and Signal Processing*, May 1989.

[Bahl 89b] Bahl, L.R., Brown, P.F., de Souza, P.V., Mercer, R.L., "A Tree-Based Statistical Language Model for Natural Language Speech Recognition", *IEEE Transactions on Acoustics, Speech and Signal Processing*, ASSP-37, No. 7, pp. 1001-1008, July 1989.

[Breiman 84] Breiman, L., Friedman, J.H., Olshen, R.A., Stone, C.j., *Classification and Regression Trees*, Wadsworth, Inc., Belmont, CA., 1984.

[Gray 82] Gray, R.M., Linde, Y., "Vector Quantizers and Predictive Quantizers for Gauss-Markov Sources", *IEEE Transactions on Communications*, Vol. COM-30, No.2, pp.381-389, February 1982.

[Hayamizu 88] Hayamizu, S., Tanaka, K., Ohta, K., "A Large Vocabulary Word Recognition System Using Rule-Based Network Representation of Acoustic Characteristic Variations", *IEEE International Conference on Acoustics Speech and Signal Processing*, pp.211-214, April 1988.

[Hayamizu 89] Hayamizu, S., Tanaka, K., Ohta, K., "On generalized description of acoustic characteristic variations of speech", *The Institute of Electronics, Information and Communication Engineers of Japan, Trans.* D, Vol.J72-D-II, No.8 [special issue on speech], pp.1215-1220, August, 1989.

[Hon 89] Hon, H.W., Lee, K.F., Weide, R., "Towards Speech Recognition Without Vocabulary-Specific Training", *Proceedings of Eurospeech*, September 1989.

[Hon 90] Hon, H.W., Lee, K.F., "On Vocabulary-Independent Speech Modeling", *IEEE International Conference on Acoustics, Speech and Signal Processing*, April, 1990.

[Huang 90] Huang, X.D., *Personal Communication*, unpublished, 1990.

[Jelinek 80] Jelinek, F., Mercer, R.L., "Interpolated Estimation of Markov Source Parameters from Sparse Data", in *Pattern Recognition in Practice*, E.S. Gelsema and L.N.Kanal ed., North-Holland Publishing Company, Amsterdam, the Netherlands, pp,381-397, 1980.

[Lee 89a] Lee, K.F., *Automatic Speech Recognition: The Development of the SPHINX System*, Kluwer Academic Publishers, Boston, 1989.

[Lee 89b] Lee, K.F., Hon, H.W., Huang, M.Y., Mahajan, S., Reddy, R., "The SPHINX Speech Recognition System", *IEEE International Conference on Acoustics, Speech and Signal Processing*, April, 1989.

[Lee 90a] Lee, K.F., Hayamizu, S., Hon, H.W., Huang, C., Swartz, J., Weide, R., "Allophone Clustering for Continuous Speech Recognition", *IEEE International Conference on Acoustics, Speech and Signal Processing*, April, 1990.

[Lee 90b] Lee, K.F., "Context-Dependent Phonetic Hidden Markov Models for Continuous Speech Recognition", *IEEE Transactions on Acoustics, Speech and Signal Processing*, April 1990.

[Linde 80] Linde, Y., Buzo, A., Gray, R.M., "An Algorithm for Vector Quantizer Design", *IEEE Transactions on Communication* COM-28, No.1, pp.84-95, January, 1980.

[Price 88] Price, P.J., Fisher, W., Bernstein, J., Pallett, D., "A Database for Continuous Speech Recognition in a 1000-Word Domain", *IEEE International Conference on Acoustics, Speech and Signal Processing*, April, 1988.

[Rabiner 89] Rabiner, L.R., Lee, C.H., Juang, B.H., Wilpon, J.G., "HMM Clustering for Connected Word Recognition", *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp.405-408, May, 1989.

[Sagayama 89] Sagayama, S., "Phoneme Environment Clustering for Speech Recognition", *IEEE International Conference on Acoustics, Speech and Signal Processing*, May, 1989.

[Schwartz 85] Schwartz, R., Chow, Y., Kimball, O., Roucos, S., Krasner, M., Makhoul, J., "Context-Dependent Modeling for Acoustic-Phonetic Recognition of Continuous Speech", *IEEE International Conference on Acoustics, Speech and Signal Processing*, April 1985.