NOTICE WARNING CONCERNING COPYRIGHT RESTRICTIONS:
The copyright law of the United States (title 17, U.S. Code) governs the making of photocopies or other reproductions of copyrighted material. Any copying of this document without permission of its author may be prohibited by law.

Speaker-Independent Phone Recognition Using Hidden Markov Models

Kai-Fu Lee and Hsiao-Wuen Hon

March 31, 1988
CMU-CS-88-121

Computer Science Department
Carnegie-Mellon University
Pittsburgh, PA 15213

Abstract

In this paper, we extend hidden Markov modeling to speaker-*independent* phone recognition. Using multiple codebooks of various LPC parameters and discrete HMMs, we obtain a speaker-independent phone recognition accuracy of 58.8% to 73.8% on the TIMIT database, depending on the type of acoustic and language models used. In comparison, the performance of expert spectrogram readers is only 69% without use of higher level knowledge. We also introduce the *co-occurrence* smoothing algorithm which enables accurate recognition even with very limited training data. Since our results were evaluated on a standard database, they can be used as benchmarks to evaluate future systems.

This research was partly sponsored by a National Science Foundation Graduate Fellowship, and by Defense Advanced Research Projects Agency Contract N00039-85-C-0163. The views and conclusions contained in this document are those of the author and should not be interpreted as representing the official policies, either expressed or implied, of the National Science Foundation, the Defense Advanced Research Projects Agency, or the US Government.

Table of Contents

1. Introduction

At present, the most popular approach in speech recognition is statistical learning, and the most successful learning technique is hidden Markov models (HMM). HMM's are capable of robust and succinct modeling of speech. Furthermore, efficient maximum-likelihood algorithms exist for HMM training and recognition. Hidden Markov models have been successfully applied to various constrained tasks, such as speaker-dependent recognition of isolated words [Averbuch 87], continuous speech [Chow 87], and phones [Schwartz 85], as well as small-vocabulary speaker-independent recognition of isolated words [Rabiner 85], and continuous speech [Rabiner 88]. In each case, extremely good results were achieved. In this study, we extend this list by applying HMM's to speaker-*independent* phone recognition in continuous speech.

There are several motivations for attempting speaker-independent phone recognition. Good phonetic decoding leads to good word decoding, and the ability to recognize the English phones accurately will undoubtedly provide the basis for an accurate word recognizer. Based on the success or failure of this study, we can predict whether large-vocabulary word recognition based on phonetic HMM's is viable. Also, by evaluating our system on a standard database, we provide a benchmark that allows direct comparison against other approaches.

One of these approaches is the knowledge engineering approach. While hidden Markov learning places learning entirely in the training algorithm, the knowledge engineering approach attempts to explicitly program human knowledge about acoustic/phonetic events into the recognizer. Whereas a HMM-based search is data-driven, a knowledge engineering search is typically heuristically guided.

After years of research, many knowledge engineering researchers [Haton 84, Cole 86a, Thompson 87] are now building speaker-independent speech recognizers using knowledge engineering techniques. These knowledge engineering techniques integrate human knowledge about acoustics and phonetics into a phone recognizer, which produces a sequence or a lattice of phones from speech signals. Currently, knowledge engineering approaches have exhibited difficulty in integrating higher level knowledge sources with the phonetic decoder. This will hopefully be overcome by more accurate phonetic decoding. It is, therefore, extremely important to evaluate the phonetic accuracy of these systems. Although different results have been published, they were based on different tasks, databases, or languages.

The recently developed TIMIT database [Lamel 86, Fisher 87] is ideal for evaluating phone recognizers. It consists of a total of 6300 sentences recorded from 630 speakers. Most of the sentences have been selected to achieve phonetic balance, and have been labeled at MIT. We will evaluate our HMM phone recognizer on this database. Our results can be used as a benchmark to evaluate other systems.

We trained phonetic hidden Markov models using 2830 TIMIT sentences from 357 speakers and tested on 160 TIMIT sentences from 20 speakers. We used multiple codebooks of

LPC-derived parameters as output observations of discrete density HMM's. Recognition was carried out by a Viterbi search that used a phone-bigram language model. With context-independent phone models, we attained a recognition rate of 64.07% for 39 English phones, and with right-context-dependent phone models, the recognition rate improved to 73.80%. We are very encouraged by this result since expert spectrogram readers at CMU are able to recognize phones without lexical knowledge with only a 69% accuracy [Weide 88]. Our results also compare well with other approaches to speaker-independent phone recognition.

We also introduce a novel smoothing algorithm, *co-occurrence smoothing*. Without smoothing, the HMM output parameters may be very sparse and some probabilities may be zeroes because the corresponding codewords were never observed. *Co-occurrence smoothing* determines the similarity between every pair of codewords from all phones, and the smooths the individual distributions accordingly. With *co-occurrence smoothing*, we are able to obtain reasonable results with only 16 sentences of training from two speakers.

In this paper, we will first describe the database and the task used in this study. Then, we will explain our HMM training and recognition algorithms. Finally, we will present results, comparisons with other speaker-independent phone recognizers, and some concluding remarks.

2. The TIMIT Database

The TIMIT (TI - MIT) acoustic/phonetic database [Lamel 86, Fisher 87], was constructed to train and evaluate speaker-independent phone recognizers. It consists of 630 speakers, each saying 10 sentences, including:

- 2 "sa" sentences, which are the same across all speakers.
- 5 "sx" sentences, which were read from a list of 450 phonetically balanced sentences selected by MIT.
- 3 "si" sentences, which were randomly selected by TI.

70% of the speakers are male. Most speakers are White adults.

We have been supplied with 19 sets of this database, where each set consists of sentences from 20 speakers, for a total of 380 speakers, or 3800 sentences. For our experiments in this study, we have designated 18 sets as training data (TID1 - TID6, TID8 - TID19) and 1 set (TID7) as testing data. We chose not to use the "sa" sentences in training or recognition, because they introduce an unfair bias for certain phones in certain contexts, which would lead to artificially high results. This leaves 2880 sentences from training, and 160 for testing. However, some of the speech data and labels were missing due to problems in reading the tape. Therefore, we actually used 2830 sentences by 357 speakers for training data, and 160 sentences by 20 speakers for test data. Table 2-1 enumerates the 160 test sentences.

SPEAKER	SENTENCES							
fdmy0	si1197-b	si567-b	si714-b	sx117-b	sx207-b	sx27-b	sx297-b	sx387-b
fjlr0	si1231-b	si1861-b	si601-b	sx151-b	sx241-b	sx331-b	sx421-b	sx61-b
fkdw0	si1207-b	si1891-b	si577-b	sx127-b	sx217-b	sx307-b	sx37-b	sx397-b
fntb0	si1203-b	si573-b	si679-b	sx123-b	sx213-b	sx303-b	sx33-b	sx393-b
fsem0	si1198-b	si1828-b	si568-b	sx118-b	sx208-b	sx28-b	sx298-b	sx388-b
futb0	si1204-b	si1330-b	si1834-b	sx124-b	sx214-b	sx304-b	sx34-b	sx394-b
mbjv0	si1247-b	si1877-b	si617-b	sx167-b	sx257-b	sx347-b	sx437-b	sx77-b
mdem0	si1868-b	si608-b	si800-b	sx158-b	sx248-b	sx338-b	sx428-b	sx68-b
mdlm0	si1234-b	si1864-b	si604-b	sx154-b	sx244-b	sx334-b	sx424-b	sx64-b
mdss0	si1881-b	si2087-b	si621-b	sx171-b	sx261-b	sx351-b	sx441-b	sx81-b
mejs0	si1240-b	si1870-b	si610-b	sx160-b	sx250-b	sx340-b	sx430-b	sx70-b
mfwk0	si1249-b	si1879-b	si619-b	sx169-b	sx259-b	sx349-b	sx439-b	sx79-b
mjee0	si1237-b	si1867-b	si607-b	sx157-b	sx247-b	sx337-b	sx427-b	sx67-b
mpam0	si1189-b	si1819-b	si1961-b	sx109-b	sx19-b	sx199-b	sx289-b	sx379-b
mpfu0	si1258-b	si1888-b	si628-b	sx178-b	sx268-b	sx358-b	sx448-b	sx88-b
mrlr0	si1196-b	si1826-b	si566-b	sx116-b	sx206-b	sx26-b	sx296-b	sx386-b
mslb0	si1193-b	si1823-b	si563-b	sx113-b	sx203-b	sx23-b	sx293-b	sx383-b
mtjs0	si1192-b	si1822-b	si562-b	sx112-b	sx202-b	sx22-b	sx292-b	sx382-b
mtkd0	si1187-b	si1817-b	si630-b	sx107-b	sx17-b	sx197-b	sx287-b	sx377-b
mtwh0	si1190-b	si1629-b	si1820-b	sx110-b	sx20-b	sx200-b	sx290-b	sx380-b

Table 2-1: List of the 160 test sentences.

There were a total of 64 possible phonetic labels. From this set, we selected 48 phones to model. We removed all "Q" (glottal stops) from the labels. We also identified 15 allophones, and folded them into the corresponding phones. Table 2-2 enumerates the list of 48 phones,

along with examples, and the allophones folded into them. Among these 48 phones, there are five groups where within-group confusions are not counted: {sil, cl, vcl, epi}, {el, l}, {en, n}, {sh, zh}, {ao, aa}, {ih, ix}, {ah, ax}. Thus, there are effectively 39 phones that are in separate categories. This folding was performed to conform to CMU/MIT standards. We found that folding closures together was necessary (and appropriate) for good performance, but folding the other categories only led to small improvements.

Phone	Example	Folded	Phone	Example	Folded
iy	*beat*		en	*button*	
ih	*bit*		ng	*sing*	eng
eh	*bet*		ch	*church*	
ae	*bat*		jh	*judge*	
ix	*roses*		dh	*they*	
ax	*the*		b	*bob*	
ah	*butt*		d	*dad*	
uw	*boot*	ux	dx	(butter)	
uh	*book*		g	*gag*	
ao	*aboutt*		p	*pop*	
aa	*cot*		t	*tot*	
ey	*bait*		k	*kick*	
ay	*bite*		z	*zoo*	
oy	*boy*		zh	*measure*	
aw	*bough*		v	*very*	
ow	*boat*		f	*fief*	
l	*led*		th	*thief*	
el	*bottle*		s	*sis*	
r	*red*		sh	*shoe*	
y	*yet*		hh	*hay*	hv
w	*wet*		cl (sil)	(unvoiced closure)	pcl,tcl,kcl,qcl
er	*bird*	axr	vcl (sil)	(voiced closure)	bcl,dcl,gcl
m	*mom*	em	epi (sil)	(epinthetic closure)	
n	*non*	nx	sil	(silence)	h#,#h,pau

Table 2-2: List of the phones used in our phone recognition task.

3. The Phone Recognizer

3.1 Speech Processing

The speech is sampled at 16 KHz, and pre-emphasized with a filter whose transfer function is $1 - 0.97z^{-1}$. Then, a Hamming widow with a width of 20 msec is applied every 10 msec. 14 LPC coefficients are computed for every 20-msec frame using the autocorrelation method. Finally, a set of 12 LPC-derived cepstral coefficients are computed from the LPC coefficients, and these LPC cepstral coefficients are transformed to a mel-scale using a bilinear transform [Shikano 85].

These 12 coefficients are vector quantized into a codebook of 256 prototype vectors of LPC cepstral coefficients. In order to incorporate additional speech parameters, we created two additional codebooks. One codebook is vector quantized from *differential coefficients*. The differential coefficient of frame n is the difference between the coefficient of frame n+2 and frame n–2. This 40-msec difference captures the slope of the spectral envelope. The other codebook is vector quantized from *energy* and *differential energy* values.

The use of multiple codebooks was first proposed by Gupta, *et al.* [Gupta 87]. We will also present comparative results using alternative methods of incorporating knowledge, such as a composite distance metric [Shikano 86a] that combines multiple feature sets in one codebook.

3.2 Context-Independent HMM Training

We first trained context-independent phonetic HMM's. One model was created for each of the 48 phones. We use the HMM topology shown in the bottom of Figure 3-1 for all 48 phones. Each phonetic HMM consists of seven states, twelve transitions, and three output probability density functions (pdf's). Each output pdf is the joint probability of the three pdf's representing the three codebooks. Thus, each HMM has 12 transition probabilities, each of which is tied to one of three sets of output pdf's, as designated on the transitions in Figure 3-1. There are a total of 256 × 3 × 3, or 2304 output probabilities. These distributions are illustrated in the upper portion of Figure 3-1, which represents a HMM for the phone /d/. The codewords for cepstrum and power are sorted by power[1]. It can be seen that first distribution has much lower power, and represents the transition from the closure into /d/. The middle distribution has higher power and shorter duration, representing the /d/ burst. The final distribution represents the transition out of /d/, and is much flatter than the other two distributions because of the variety of contexts it absorbed. This /d/ model is well-trained and robust, as evidenced by the scarcity of zero probabilities in the output pdf's.

---

[1]Although power was not used in the cepstrum codebook, the power value for each cepstral vector was carried along in the codebook generation process for the purpose of sorting the codewords.

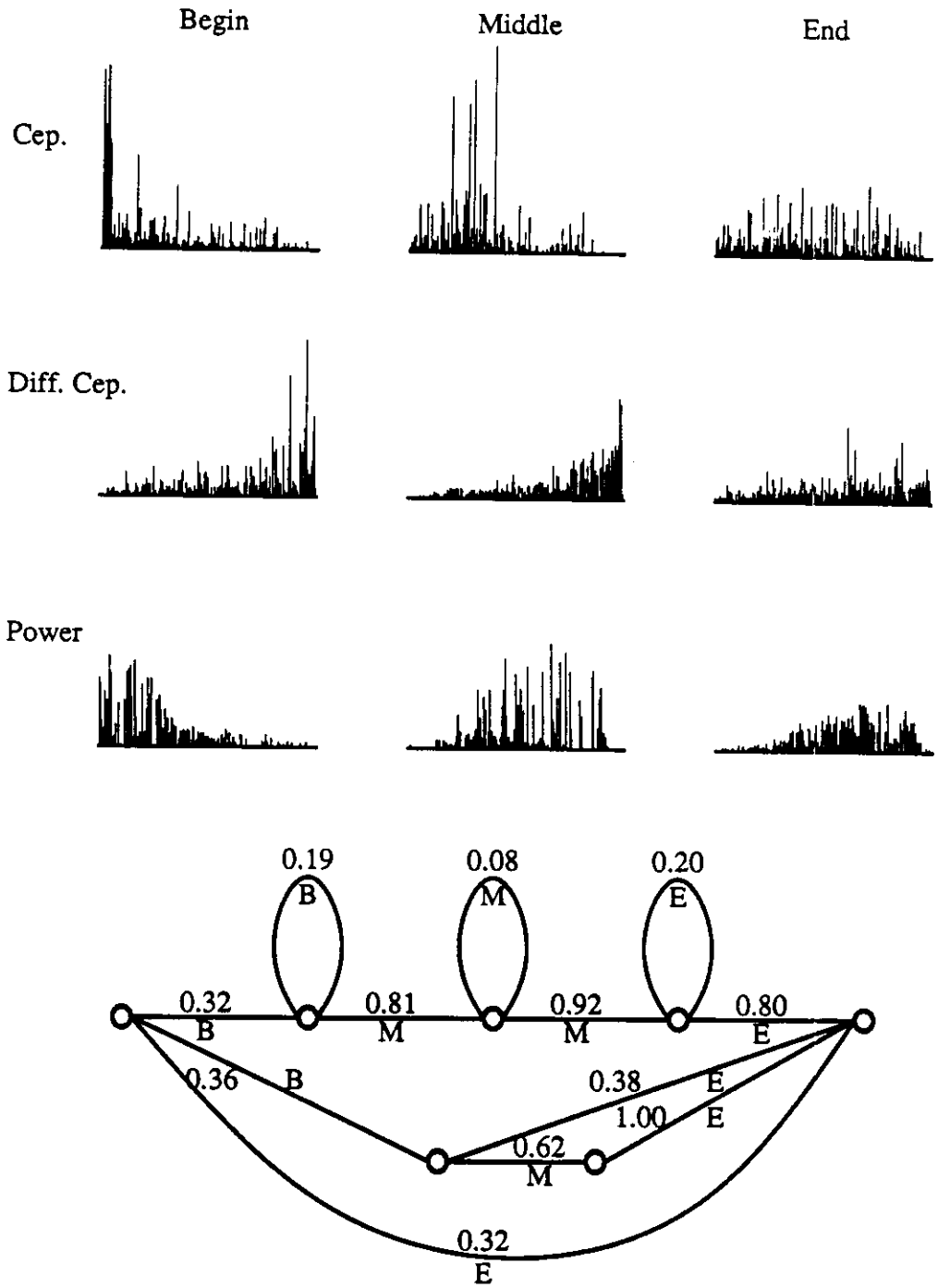

Figure 3-1: A phonetic hidden Markov model for phone /d/. The upper portion displays the 9 output pdf's, where the x-axis is the codeword index, sorted by power, and the y-axis is probability. The lower portions shows the HMM topology with transition probabilities. Transitions with the same label are tied to the same output pdf.

Uniform initialization is used for the transition probabilities, i.e., all transitions from a state are considered equally likely initially. The output probabilities are initialized from the segmented and labeled TIMIT sentences. For each codebook, a histogram for all codewords is accumulated and then normalized for each phone. All three distributions are initialized with the same normalized codebook histogram. This technique was first used by Schwartz, *et al.* [Schwartz 85].

Then, we ran three iterations of the forward-backward algorithm [Jelinek 76] over all training sentences. For each training sentence, we used the labels provided by MIT, but not the boundaries. The sequence of HMM's corresponding to the TIMIT phone labels are concatenated into a large sentence HMM, and forward-backward algorithm is run on the entire sentence HMM. After each iteration over all the sentences, the parameters are re-estimated.

Finally, the output parameters are smoothed using a novel smoothing technique, *co-occurrence* smoothing. We define $CP(i \mid j)$, the *co-occurrence probability* of codeword i given codeword j, as[2]:

$$CP(i \mid j) = \frac{\sum_{p=1}^{NP} \sum_{d=1}^{ND(p)} P(i \mid p,d) \cdot P(j \mid p,d) \cdot P(p) \cdot P(d)}{\sum_{k=1}^{NC} \sum_{p=1}^{NP} \sum_{d=1}^{ND(p)} P(k \mid p,d) \cdot P(j \mid p,d) \cdot P(p) \cdot P(d)} \tag{1}$$

where NP is the number of phones, $ND(p)$ is the number of output pdf's in the HMM for phone p, NC is the number of codewords in the codebook, and $P(k \mid p,d)$ is the output probability of codeword k for distribution d in phone model p. *Co-occurrence probability* can be loosely defined as "when codeword j is observed, how often is codeword i observed in similar contexts". In our definition, "similar context" means the same output pdf.

If $P(k \mid p,d)$, the output probabilities, are under-trained, as often is the case, the distributions will be sharp and many zeroes will be present. This will lead to poor results in recognition. We could use the *co-occurrence probability* (CP) to smooth the output pdf's (P) into a smoothed pdf (SP):

$$SP(k \mid p,d) = \sum_{i=1}^{NC} CP(k \mid i) \cdot P(i \mid p,d) \tag{2}$$

Although $SP(k \mid p,d)$ does not suffer from sparseness, it may be too smooth. Therefore, a compromise can be reached by combining the two pdf's:

$$MP(k \mid p,d) = \lambda_c \cdot P(k \mid p,d) + (1-\lambda_c) \cdot SP(k \mid p,d) \tag{3}$$

λ_c depends on c, the count of the distribution being smoothed. A larger c implies that $P(k \mid p,d)$

[2]The *co-occurrence probabilities* can be more conveniently computed from the counts accumulated in forward-backward by a simple transformation of the equation.

is reliable, and suggests a larger λ_c. λ_c can be automatically estimated using deleted interpolation [Jelinek 80]. In our implementation, λ_c is estimated by running 100 iterations of deleted interpolation smoothing. A λ_c is estimated not for a particular count, but for a range of counts. Figure 3-2 shows the effect of smoothing on a poorly trained pdf.

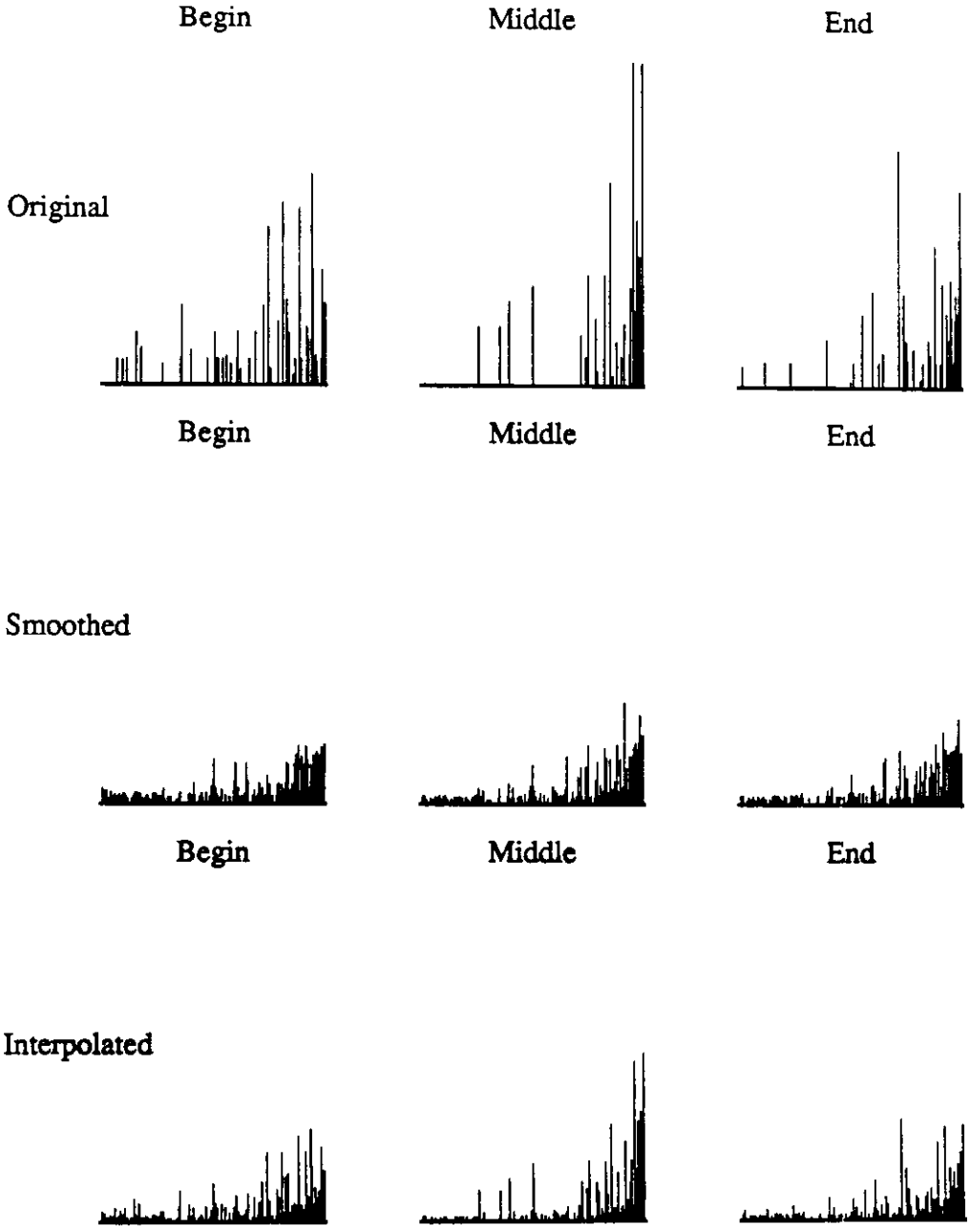

Figure 3-2: The effect of co-occurrence smoothing and deleted interpolation. The top pdf's represent the cepstrum codebook of the unsmoothed model for /ae/ (*P*), which was purposely undertrained. The second set of pdf's has been smoothed (*SP*). The third set represents the interpolated pdf's (*MP*)

Co-occurrence smoothing is an extension of the *correspondence* smoothing proposed by Sugawara, *et al.* [Sugawara 86]. *Correspondence* smoothing counts the frequency that two codewords are aligned against each other in DP matching between the same words. These counts are then normalized into a probabilistic mapping like *CP*, and are used to smooth the output pdf's. *Co-occurrence* smoothing is similar in that it measures the likelihood that two labels will occur in similar contexts, except it has several additional advantages:

1. *Co-occurrence* smoothing works on continuous speech and does not require segmentation.
2. *Co-occurrence* smoothing is text-independent; it is not necessary to say any fixed training text for smoothing.
3. *Co-occurrence* smoothing operates directly on the output pdf's, and does not require DP.
4. *Co-occurrence* smoothing is more relaxed than correspondence smoothing. Two codes don't have to be exactly aligned to train the *CP* matrix. So less training data is needed.
5. Our use of deleted interpolation to combine the two matrices gives superior results than using either one alone.

For context-independent HMM training, we divided the training data into two blocks during the final iteration of Forward-Backward, and then trained λ's to interpolate: (1) HMM parameters, (2) co-occurrence smoothed HMM parameters, and (3) uniform distribution. A λ was used for each pre-determined range of HMM parameter counts.

3.3 Context-Dependent HMM Training

Context-independent phone models assume that speech is produced as a sequence of concatenated phones, which are unaffected by context. While we may attempt to produce speech in such a manner, our articulators cannot move instantaneously from one location for one phone to another for the next phone. Thus, in reality, phones are highly affected by the neighboring phonetic contexts.

Context-independent models attempt to account for this effect by making the begin and end pdf's flatter, thereby increasing the weight for the stationary middle pdf. However, useful information in the boundary pdf's is destroyed by combining all contexts together.

A context-dependent phone model [Schwartz 85] is one that is dependent on the left and/or right neighboring phone. With N phones, there are potentially N^2 context-dependent phones if we model left or right context, and N^3 if we model both. We cannot hope to adequately train so many models. Fortunately, since we use phone models, we always have the better trained, but less accurate context-independent phone models. By interpolating the two, we will have models that are better trained than the context-dependent models, and more accurate than the context-independent ones. Again, we can use deleted interpolation to combine the two estimates.

In our implementation, we use right-context dependent phone modeling. For example, the sentence `/sil hh ix dx en sil m ...../` would be transformed into `/sil(hh) hh(ix) ix(dx) dx(en) en(sil) sil(m) ...../`, where `x(y)` designates phone `x` with right context `y`. There were a total of 1450 right-context-dependent models.

The context-dependent HMM's were initialized with statistics from the corresponding context-independent HMM's. We ran two iterations of context-dependent. forward-backward training. During the last iteration, training data were divided into two blocks, and context-independent and context-dependent counts were maintained for each block. Context-independent counts were obtained by adding together all corresponding right-context-dependent models of the phone. After these two iterations, deleted interpolation was used to interpolate: (1) right-context-dependent model parameters, (2) context-independent model parameters, (3) co-occurrence smoothed context-dependent model parameters, and (4) a uniform distribution. These interpolated context-dependent models were then used for recognition.

3.4 HMM Recognition

Recognition is carried out by a Viterbi search [Viterbi 67] in a large HMM. For context-independent phone recognition, an initial and a final state are created. The initial state is connected with null arcs to the initial state of each phonetic HMM, and null arcs connect the final state of each phonetic HMM to the final state. The final state is also connected to the initial state. This HMM is illustrated in Figure 3-3(a).

For context-dependent phone models, each right-context-dependent model is only connected to successors that correspond to the appropriate right phone context. However, some legal right contexts are not covered in the training data, and no corresponding right-context model was created. Therefore, for all unobserved right contexts, we connect the context-independent models to them. As a result, the network has one and only one phone-level path for any sequence of phones. This HMM is illustrated in Figure 3-3(b).

The Viterbi search finds the optimal state sequence in this large HMM. At each time frame, the data structures are updated by finding the path with the highest probability to each state at this time. When the entire sentence has been consumed, a backtrace recovers the state sequence, which uniquely identifies the recognized phone sequence. Since the number of states is moderate, a full search is possible.

The HMM's were trained to maximize $P(Observations \mid Model)$, while in recognition we need $P(Model \mid Observations)$. By Bayes' rule,

$$P(Model \mid Observation) = \frac{P(Observations \mid Model) \cdot P(Model)}{P(Observation)} \tag{4}$$

Since the *Observation* is given, $P(Observation)$ is a constant, and only the numerator need be evaluated. To evaluate the numerator, we need $P(Model)$, or a language model, in recognition.

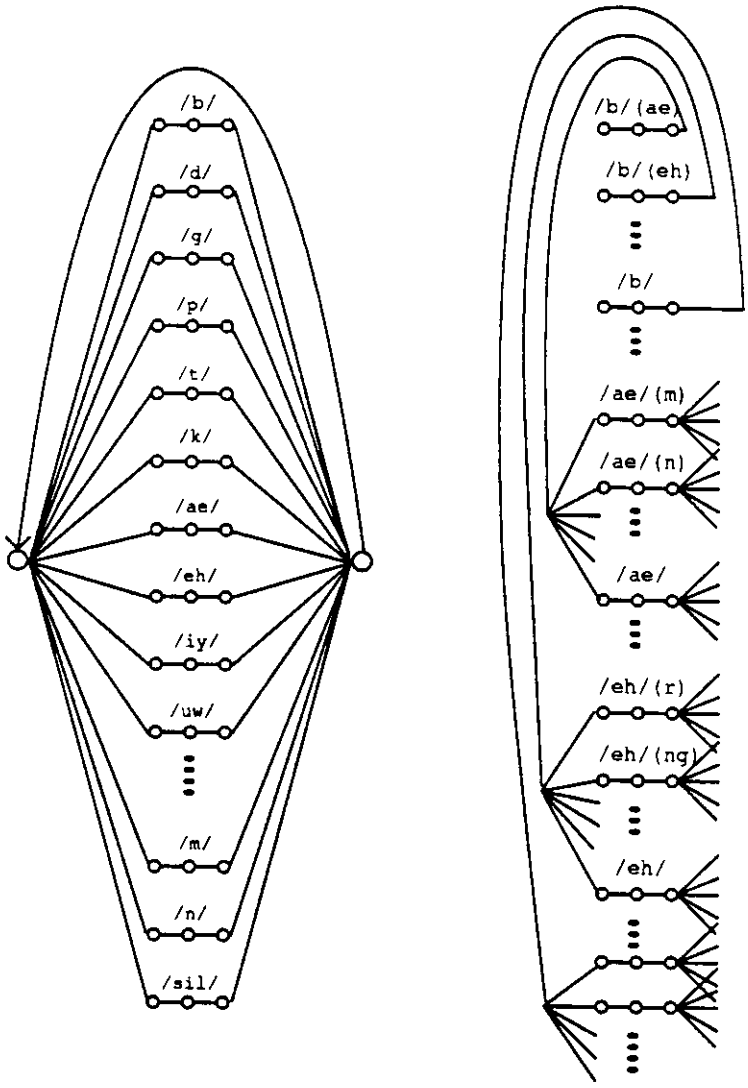

Figure 3-3: HMM's used for context-independent (a) and right-context-dependent phone recognition (b). For right-context-dependent phone recognition, `/X/(Y)` designates phone `/X/` in the right context of `/Y/`

This probability would be multiplied by the acoustic probability every time a phone transition occurs. In this study, we use a bigram phone language model that estimates the probability of a phone given the previous phone. This bigram was estimated from the same TIMIT training set. This is the same language model used by [Schwartz 85].

4. Results and Discussion

4.1 Phone Recognition Results

We tested our phone recognizer on the TIMIT database. As described in Chapter 2, we used 18 sets (2830 sentences by 357 speakers) to train our HMM's, and one set (TID7) (160 sentences by 20 speakers) to test our system. Our phone recognition results are shown in Table 4-1. With context-independent phone modeling, out of a total of 6061 phones, 3883 were correctly identified for a phone recognition accuracy of 64.07%. With right-context-dependent phone modeling, 4473 were correctly recognized, increasing the recognition accuracy to 73.80%. The number of correct, substituted, inserted, and deleted phones are computed by a DP match between the correct phone string and the recognized phone string. Substitutions within the following sets are not counted as errors: {sil, cl, vcl, epi}, {el, l}, {en, n}, {sh, zh}, {ao, aa}, {ih, ix}, {ah, ax}. Recognition rates for four broad classes (sonorant, stop, fricative, and closure) are reported in Table 4-2. For these experiments, a bigram phone-language model is used, insertions are *not* counted as errors, and are held to less than 12% by appropriately weighing the language model and acoustic model probabilities. These conditions are identical to that used by Schwartz, *et al.* [Schwartz 85].

	Context-Indep.	Context-Dep.
Correct	64.07% (3883)	73.80% (4473)
Substitutions	26.22% (1589)	19.62% (1189)
Deletions	9.72% (589)	6.58% (399)
Insertions	10.79% (654)	7.72% (468)

Table 4-1: Phone recognition results with context-independent and context-dependent models.

Class	Occurrences	Context-Indep.	Context-Dep.
Sonorant	3027	53.68% (1625)	65.71% (1989)
Stop	1014	58.09% (589)	69.92% (709)
Fricative	736	66.03% (486)	78.40% (577)
Closure	1284	92.13% (1183))	93.30% (1198)

Table 4-2: Phone recognition results by broad phone class.

4.2 Additional Experiments

4.2.1 The Phone Language Model

One question that can be raised is the validity of using a bigram language model. We believe that for some comparisons, it is certainly valid. For example, since Schwartz, *et al.* [Schwartz 85] used exactly the same model for speaker-dependent recognition, that comparison is clearly valid. We believe the bigram model is also fair when comparing against expert spectrogram readers, because they have far more knowledge about the likelihood of various combinations of phonetic events than bigrams. Also, they may subconsciouly use their lexical knowledge in spite of their attempt to suppress it.

On the other hand, systems that use Bayesian classification implicitly assume the *a priori* distribution of the phones, which is equivalent to a *unigram* model. Some other systems might use no language model at all, or the *zero-gram* model. In order to validate these comparisons, we ran our system with bigram, unigram, and zero-gram laguage models, and present our results in Table 4-3. As expected, the use of simpler language models led to some degradations.

Language Model	Context-Independent Recognition Rate	Context-Dependent Recognition Rate
Bigram	64.07%	73.80%
Unigram	60.91%	70.38%
None	58.77%	69.51%

Table 4-3: Phone recognition results with different phone language models.

4.2.2 Utility of Additional Features and Codebooks

In order to evaluate the utility and to justify the overhead of multiple codebooks and additional features, we ran a set of experiments where we used various combinations of the features with varying numbers of codebooks. Table 4-4 shows the results for context-independent phone modeling. To use multiple feature sets in a codebook, inter-set distances are computed, and combined using a linear combination [Shikano 85]. The weights in the linear combination are optimized using a separate set of tuning sentences.

We find that using only one set of the features in one codebook produced poor results. Linearly combining all three sets of a features in one codebook [Furui 86, Shikano 86b] led to a much better result. However, equivalent results could be obtained by discarding a feature set and adding a codebook, and much better results can be obtained by using all three sets of features in three individual codebooks.

4.2.3 Utility of Co-occurrence Smoothing

All of the above results were obtained with *co-occurrence* smoothing and deleted interpolation. We also tested our recognizer with no smoothing, and with floor smoothing (replacing all probabilities smaller than the floor with the floor). We used context-independent

Cep.	DCep.	Pow.	Codebooks	Recog. Rate
X			1	49.78%
	X		1	46.11%
		X	1	31.91%
X	X	X	1	58.62%
X	X		2	57.93%
X		X	2	57.99%
	X	X	2	55.01%
X	X	X	3	64.07%

Table 4-4: Phone recognition results using context-*independent* phone models, and various combinations of features and number of codebooks.

phone models for this experiment. We found that 2830 training sentences, the results were not significantly different because the context-independent HMM's were very well trained, and did not require smoothing. To test whether *co-occurrence* smoothing would help when the amount of training data is inadequate, we ran a set of experiments where we reduced the amount of training data. The context-independent phone recognition results are shown in Figure 4-1. This illustrates that when we have insufficient training data, smoothing can result in dramatic improvements. Also, we see that *co-occurrence* smoothing is significantly better than floor smoothing. For example, the results from *co-occurrence* smoothing on two speakers is equivalent to floor smoothing on five or no smoothing on 15.

4.3 Discussion

Without using lexical or higher level knowledge, expert spectrogram readers could recognize phones from continuous speech with an accuracy of 69% [Weide 88]. Our HMM recognizer is already beyond that level of performance. This suggests that any approach that solely emulates spectrogram reading is unlikely to produce better results than those presented here. This is not to say that knowledge engineering approaches cannot do better, because the expert spectrogram readers evaluated are not as good as Victor Zue [Cole 80], and because spectrogram reading is only one of many kinds of human perceptual and speech knowledge.

Comparison against other speaker-independent recognizers is more difficult, because of the different databases, training data, phone classes, and additional information used. For example, vowel recognition is considerably harder than phone recognition. The use of hand-segmentation eliminates the possibility of deletions and insertions, thereby increases recognition accuracy. It was precisely because of this lack of uniformity that we believe a our benchmark result would be useful. With this in mind, we now present the results of several other systems.

The ANGEL Acoustic-Phonetic Group at CMU has been working on speaker-independent

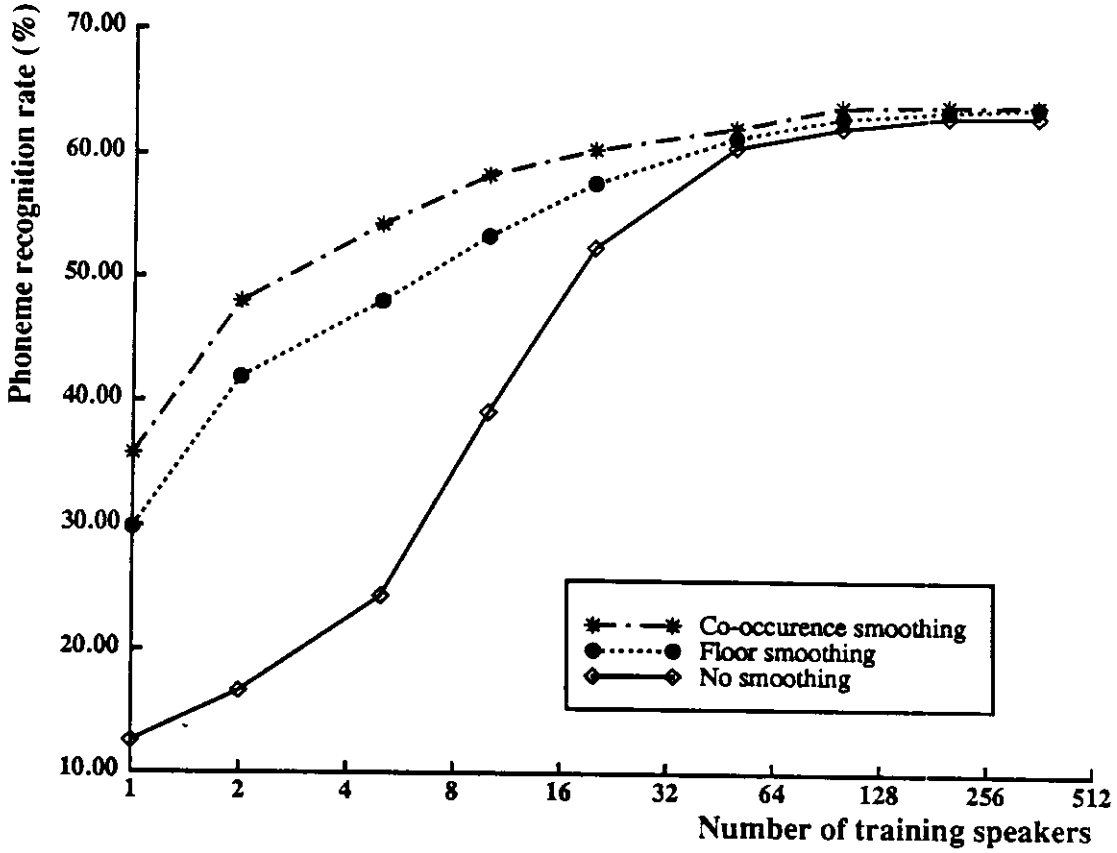

Figure 4-1: Phone recognition results with context-independent models, using different number of training speakers and smoothing algorithms.

phone recognition for several years. Their earlier results can be found in [Cole 86b]. The current recognition accuracy has been improved. On the same test set using a unigram phone language model, the ANGEL System achieved an accuracy of 55% [Chigier 88].

Nakagawa [Nakagawa 86] applied statistical pattern classification and dynamic time warp to speaker-independent phone recognition. He reported 51% and 56% with these two approaches for 7 vowels, 71% and 74% for 9 fricatives, and 57% and 55% for 9 stops and nasals. For this experiment, hand segmentation was used, and different classes were evaluated separately so that no between-class confusions could occur. No language model was used in this task.

Leung and Zue [Leung 88] used artificial neural networks for the recognition of 16 vowels, and reported 54% for context-independent recognition, and 67% for context-dependent. In this experiment, hand segmentation was used for training and testing. The correct context was provided for both training and recognition using context-dependent networks.

Another interesting comparison is BBN's speaker-*dependent* phone recognizer [Schwartz 85]. They reported phone recognition rates of 61% and 79% for one very good speaker using context-independent and left-context-dependent models, respectively [Schwartz 85]. Our results

for speaker-*independent* phone recognition are not far from the BBN speaker-*dependent* results. This was made possible by several factors: (1) we benefited from much more training data that is available for speaker-*independent* tasks, (2) differential and power information are very useful for speaker-independent recognition, and (3) the use of multiple codebooks was a good way to combine multiple feature sets. With context-independent phone models, our results are actually significantly better. However, when context-dependent models were added, our improvement was much smaller. We attribute this to our use of differential parameters, which already accounted for some contexual variations by emphasizing stationary portions of phones.

In spite of the high accuracy we achieved, we see many areas where we might get further improvements: (1) increase the amount of training, (2) modeling of left *and* right context [Schwartz 85], (3) use of continuous parameters [Brown 87, Rabiner 88], (4) use of maximum mutual information estimation [Brown 87], and (5) incorporation of additional knowledge sources, such as duration, or the output of a knowledge-based phone decoder. However, having demonstrated the feasibility of speaker-independent phone recognition, our future work will focus on the creation of a large-vocabulary speaker-independent continuous speech recognition system based on the methods used in this study.

5. Conclusion

In this paper, we extended the currently popular hidden Markov modeling technique to speaker-independent phone recognition. This is the first time that HMM has been applied to this task. Using multiple codebooks of LPC-derived parameters, discrete HMM, and Viterbi decoding, we obtained a 73.80% speaker-independent phone recognition accuracy in continuous speech. Moreover, by using a novel smoothing technique, *co-occurrence* smoothing, we were able to get very respectable results from just a few training speakers. Our results are the best reported thus far on this database.

We used the TIMIT database for evaluating our recognizer. This allows other researchers to evaluate their techniques on the same training and testing data. We believe this benchmark result will prove useful to other researchers, especially those using knowledge based approaches.

We began this study with the hope of building a successful phone recognizer that could provide the basis for a speaker-independent continuous speech recognizer. We have shown good recognition results can be obtained for speaker-independent phone recognition, and are now working to extend this work to a large-vocabulary speaker-independent continuous speech recognition system [Lee 88].

Acknowledgments

The authors wish to thank Raj Reddy for his motivation and guidance; Kiyohiro Shikano for providing the LPC routines; and Ron Cole, Richard Stern, and Victor Zue for reading drafts of this paper.

References

[Averbuch 87] Averbuch, et al.
Experiments with the Tangora 20,000 Word Speech Recognizer.
In *IEEE International Conference on Acoustics, Speech, and Signal Processing*. April, 1987.

[Brown 87] Brown, P.
The Acoustic-Modeling Problem in Automatic Speech Recognition.
PhD thesis, Carnegie-Mellon University, May, 1987.

[Chigier 88] Chigier, B.
Personal Communication.
unpublished.
1988

[Chow 87] Chow, Y.L., Dunham, M.O., Kimball, O.A., Krasner, M.A., Kubala, G.F., Makhoul, J., Roucos, S., Schwartz, R.M.
BYBLOS: The BBN Continuous Speech Recognition System.
In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 89-92. April, 1987.

[Cole 80] Cole, R. A., Rudnicky, A. I., Zue, V. W., Reddy, D. R.
Speech as Patterns on Paper.
In R. A. Cole (editor), *Perception and Production of Fluent Speech.* Lawrence Erlbaum Associates, Hillsdale, N.J., 1980.

[Cole 86a] Cole, R. A., Phillips, M., Brennan, B., Chigier, B.
The C-MU Phonetic Classification System.
In *IEEE International Conference on Acoustics, Speech, and Signal Processing*. April, 1986.

[Cole 86b] Cole, R. A.
Phonetic Classification in New Generation Speech Recognition Systems.
In *Speech Tech. 86*, pages 43-46. 1986.

[Fisher 87] Fisher, W.M., Zue, V., Bernstein, J., Pallett, D.
An Acoustic-Phonetic Data Base.
In *113th Meeting of the Acoustical Society of America.* May, 1987.

[Furui 86] Furui, S.
Speaker-Independent Isolated Word Recognition Using Dynamic Features of Speech Spectrum.
IEEE Transactions on Acoustics, Speech, and Signal Processing
ASSP-34(1):52-59, February, 1986.

[Gupta 87] Gupta, V.N., Lennig, M., Mermelstein, P.
Integration of Acoustic Information in a Large Vocabulary Word Recognizer.
In *IEEE International Conference on Acoustics, Speech, and Signal Processing*, pages 697-700. April, 1987.

[Haton 84] Haton, J.-P.
Knowledge-based and Expert Systems in Automatic Speech Recognition.
In DeMori, R. (editor), *New Systems and Architectures for Automatic Speech Recognition and Synthesis*. Dordrecht, Reidel, Netherlands, 1984.

[Jelinek 76] Jelinek, F.
Continuous Speech Recognition by Statistical Methods.
Proceedings of the IEEE 64(4):532-556, April, 1976.

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

[Lamel 86] Lamel, L.F., Kassel, R.H., Seneff, S.
Speech Database Development: Design and Analysis of the Acoustic-Phonetic Corpus.
In Baumann, L.S. (editor), *Proceedings of the DARPA Speech Recognition Workshop*, pages 100-109. February, 1986.

[Lee 88] Lee, K.F.
Large-Vocabulary Speaker-Independent Continuous Speech Recognition: The SPHINX System.
PhD thesis, Carnegie-Mellon University, 1988.

[Leung 88] Leung, H.C., Zue, V.W.
Some Phonetic Recognition Experiments Using Artificial Neural Nets.
In *IEEE International Conference on Acoustics, Speech, and Signal Processing*. April, 1988.

[Nakagawa 86] Nakagawa, S.
Speaker-Independent Phoneme Recognition in Continuous Speech by a Statistical Method and a Stochastic Dynamic Time Warping Method.
Technical Report CMU-CS-86-102, Carnegie-Mellon University, January, 1986.

[Rabiner 85] Rabiner, L. R., Juang, B. H., Levinson, S. E., Sondhi, M. M.
Recognition of Isolated Digits Using Hidden Markov Models With Continuous Mixture Densities.
AT&T Technical Journal 64(6):1211-33, July-August, 1985.

[Rabiner 88] Rabiner, L.R., Wilpon, J.G., Soong, F.K.
High Performance Connected Digit Recognition Using Hidden Markov Models.
In *IEEE International Conference on Acoustics, Speech, and Signal Processing*. April, 1988.

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

[Shikano 85] Shikano, K.
Evaluation of LPC Spectral Matching Measures for Phonetic Unit Recognition.
Technical Report, Carnegie-Mellon University, May, 1985.

[Shikano 86a] Shikano, K., Lee. K, Reddy, D. R.
Speaker Adaptation through Vector Quantization.
In *IEEE International Conference on Acoustics, Speech, and Signal Processing*. April, 1986.

[Shikano 86b] Shikano, K.
Evaluation of LPC Spectral Matching Measures for Phonetic Unit Recognition.
Technical Report, Carnegie-Mellon University, 1986.

[Sugawara 86] Sugawara, K., Nishimura, M., Kuroda, A.
Speaker Adaptation for a Hidden Markov Model.
In *IEEE International Conference on Acoustics, Speech, and Signal Processing*. April, 1986.

[Thompson 87] Thompson, H.S., Laver, J.D.
The Alvey Speech Demonstrator - Architecture, Methodology, and Progress to Date.
In *Proceedings of Speech Tech.* 1987.

[Viterbi 67] Viterbi, A. J.
Error Bounds for Convolutional Codes and an Asymptotically Optimum Decoding Algorithm.
IEEE Transactions on Information Theory IT-13(2):260-269, April, 1967.

[Weide 88] Weide, R.
Personal Communication.
unpublished.
1988