# Speaker-Independent Phone Recognition Using Hidden Markov Models

KAI-FU LEE, MEMBER, IEEE, AND HSIAO-WUEN HON

***Abstract*—In this paper, we extend hidden Markov modeling to speaker-*independent* phone recognition. Using multiple codebooks of various LPC parameters and discrete HMM's, we obtain a speaker-independent phone recognition accuracy of 58.8–73.8 percent 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 percent 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.**

## I. 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 [1], continuous speech [2], and phones [3], as well as small-vocabulary speaker-independent recognition of isolated words [4], and continuous speech [5]. 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 an HMM-based search is data driven, a knowledge engineering search is typically heuristically guided.

After years of research, many knowledge engineering researchers [6]–[8] 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 [9], [10] 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 percent for 39 English phones, and with right-context-dependent phone models, the recognition rate improved to 73.80 percent. 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 percent accuracy [11]. Our results also compare well to other approaches to speaker-independent phone recognition.

We also introduce a novel smoothing algorithm, *co-occurrence smoothing*. Without smoothing, the HMM out-

Manuscript received April 11, 1988; revised February 4, 1989. This work was supported in part by a National Science Foundation Graduate Fellowship, and by Defense Advanced Research Projects Agency Contract N00039-85-C-0163. The views and conclusions contained herein are those of the authors 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 U.S. Government.

The authors are with the Computer Science Department, Carnegie-Mellon University, Pittsburgh, PA 15213.

IEEE Log Number 8930533.

0096-3518/89/1100-1641$01.00 © 1989 IEEE

put parameters may be very sparse and some probabilities may be zero because the corresponding codewords were never observed. *Co-occurrence smoothing* determines the similarity between every pair of codewords from all phones, and then 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 to other speaker-independent phone recognizers, and some concluding remarks.

## II. The TIMIT Database

The TIMIT (TI-MIT) acoustic/phonetic database [9], [10], 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.

Seventy percent 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.

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 I enumerates the list of 48 phones, along with examples, and the allophones folded into them. Among these 48 phones, there are seven 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.

TABLE I
List of the Phones Used in Our Phone Recognition Task

| 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 | *about* | | 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 |

## III. The Phone Recognizer

### A. Speech Processing

The speech is sampled at 16 kHz, and preemphasized with a filter whose transfer function is $1\text{-}0.97z^{-1}$. Then, a Hamming window with a width of 20 ms is applied every 10 ms. Fourteen LPC coefficients are computed for every 20-ms 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 [12], [13].

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-ms difference captures the slope of the spectral envelope. The other codebook is vector quantized from *log power* and *differential log power* values. For normalization, the maximum log power is subtracted from each log power value.

The use of multiple codebooks was first proposed by Gupta *et al.* [14]. Multiple codebooks reduce the VQ distortion and increase the dynamic range of the system; [14] and [15] contain more detailed analysis of the use of multiple codebooks. We will also present comparative results using alternative methods of incorporating knowledge, such as a composite distance metric [16] that combines multiple feature sets in one codebook.

### *B. Context-Independent HMM Training*

We first trained context-independent phonetic HMM's. One model was created for each of the 48 phones. We tested many HMM topologies, and found the one shown in the bottom of Fig. 1 to be the best. Each of the 48 phones is represented by an HMM that 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. By assuming independence of the three codebooks, the output probability can be computed as the product of probabilities of three codewords from the three codebooks. Thus, each HMM has 12 transition probabilities, each of which is tied to one of three sets of output pdf's (B—begin, M—middle, E—end) as designated on the transitions in Fig. 1. There are a total of 256 × 3 × 3, or 2304 output parameters for each HMM. These distributions are illustrated in the upper portion of Fig. 1, which represents an HMM for the phone /d/. The codewords for cepstrum and power are sorted by power.[1] It can be seen that the 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. Because of plentiful training data, this /d/ model is robust, as evidenced by the scarcity of zero probabilities in the output pdf's.

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.* [3].

Three iterations of the forward-backward algorithm [17] were run over all training sentences. For each training sentence, we used the labels provided by MIT, but not the boundaries. Thus, the removal of glottal stops did not present any problems. The sequence of HMM's corresponding to the TIMIT phone labels are concatenated into a large sentence HMM, and a forward-backward algorithm is run on the entire sentence HMM. After each iteration over all the sentences, the parameters are reestimated.

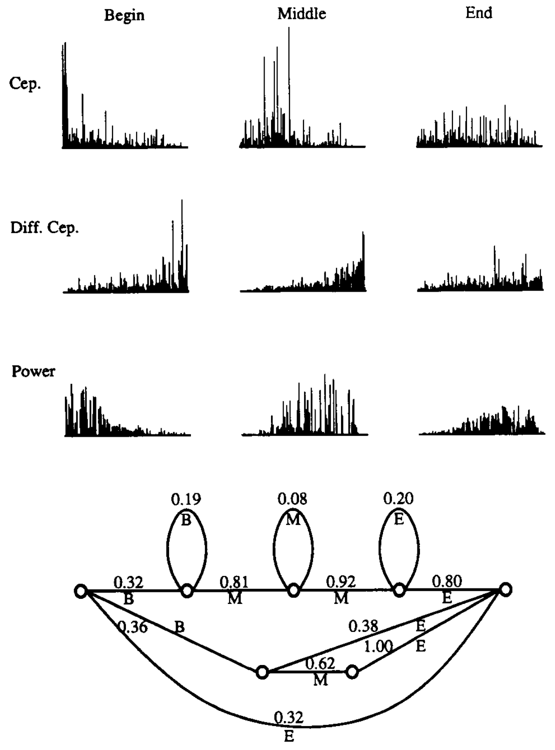

Fig. 1. A phonetic hidden Markov model for phone /d/. The upper portion displays the 9 output pdf's, where the $x$-axis is the codeword index, sorted by power, and the $y$-axis is probability. The lower portion shows the HMM topology with transition probabilities. Transition labels B, M, E, represent the beginning, middle, and ending output pdf's, respectively. Transitions with the same label are tied to the same output pdf.

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

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

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 $P(k\,|\,p, d)$ is the output probability of codeword $k$ for distribution $d$ in phone model $p$. With context-independent phones, $NC = 256$, $NP = 48$, and $ND(p) = 3$ for all $p$. *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

[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.

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

same output pdf. A similar smoothing technique was used in [18].

If $P(k|p, d)$, the output probabilities, are undertrained, as often is the case, the distributions will be sharp and many zeros 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 ($SP$):

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

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

$$MP(k|p, d) = \lambda_c \cdot P(k|p, d) + (1 - \lambda_c) \cdot SP(k|p, d). \quad (3)$$

$\lambda_c$ depends on $c$, the count of the distribution being smoothed. A larger $c$ implies that $P(k|p, d)$ is reliable, and suggests a larger $\lambda_c$. $\lambda_c$ can be automatically estimated using deleted interpolation [19]. In our implementation, $\lambda_c$ is estimated by running 100 iterations of deleted interpolation smoothing. A $\lambda_c$ is estimated not for a particular count, but for a range of counts. Fig. 2 shows the effect of smoothing on a poorly trained pdf.

*Co-occurrence* smoothing is an extension of the *correspondence* smoothing proposed by Sugawara *et al.* [20]. *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 the following several additional advantages:

1) *co-occurrence* smoothing works on continuous speech and does not require segmentation;

2) *co-occurrence* smoothing operates directly on the output pdf's, and does not require DP; and

3) *co-occurrence* smoothing is more relaxed than correspondence smoothing. Two codes do not have to be exactly aligned to train the *CP* matrix. So fewer training data are needed.

In addition, *co-occurrence* smoothing is text-independent; it is not necessary to speak any fixed training text for smoothing. While this might compromise some accuracy, it is more convenient and transparent to the user. Finally, our use of deleted interpolation is useful in avoiding oversmoothing.

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 predetermined range of HMM parameter counts.

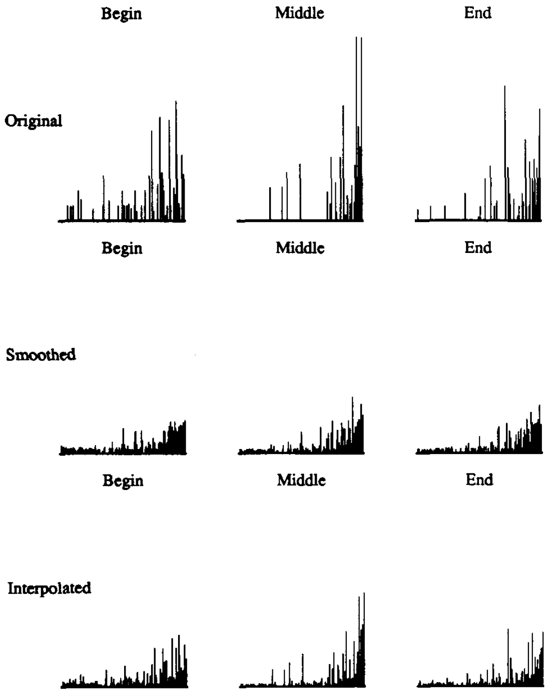

Fig. 2. The effect of co-occurrence smoothing and deleted interpolation. The top pdf's represent the cepstrum codebook of the unsmoothed model for /ae/ ($P$), which was purposely undertrained. The second set of pdf's has been smoothed ($SP$). The third set represents the interpolated pdf's ($MP$).

### C. 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 [3] 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. Our choice of right-context dependent model was guided by the fact that most phonemes are affected by both left and right contexts, but prevocalic stops are affected more by the right context.

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.

### D. HMM Recognition

Recognition is carried out by a Viterbi search [21] 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 Fig. 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 sequences of phones. This HMM is illustrated in Fig. 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.

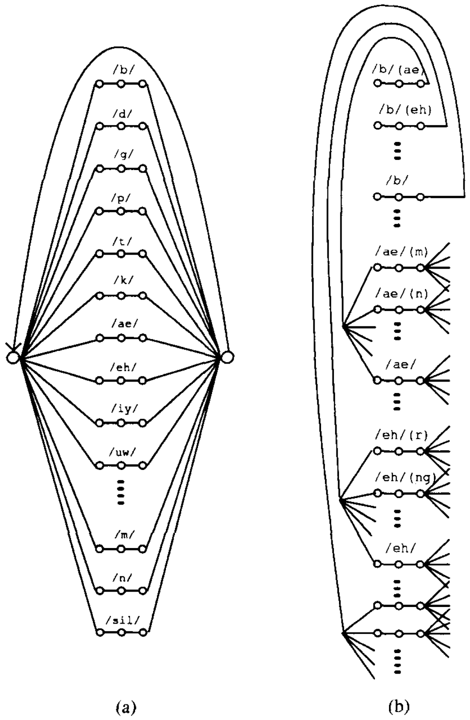

Fig. 3. HMM's used for context-independent (a) and right-context-dependent phone recognition (b). For right-context-dependent phone recognition, /X/ (Y) designates phone /X/ in the right context of /Y/.

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

$$P(\mathit{Model} \mid \mathit{Observation}) = \frac{P(\mathit{Observations} \mid \mathit{Model}) \cdot P(\mathit{Model})}{P(\mathit{Observation})}. \quad (4)$$

Since the *Observation* is given, $P(\mathit{Observation})$ is a constant, and only the numerator need be evaluated. To evaluate the numerator, we need $P(\mathit{Model})$, or a language model, in recognition. 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 [3].

## IV. Results and Discussion

### A. Phone Recognition Results

We tested our phone recognizer on the TIMIT database. As described in Section II, 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 II. With context-independent phone modeling, out of a total of 6061 phones, 3883 were correctly identified for a phone

TABLE II
PHONE RECOGNITION RESULTS WITH CONTEXT-INDEPENDENT AND CONTEXT-DEPENDENT MODELS

| | 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 III
PHONE RECOGNITION RESULTS BY BROAD PHONE CLASS

| 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 IV
PHONE RECOGNITION RESULTS WITH DIFFERENT PHONE LANGUAGE MODELS

| 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 V
PHONE RECOGNITION RESULTS USING CONTEXT-*INDEPENDENT* PHONE MODELS, AND VARIOUS COMBINATIONS OF FEATURES AND NUMBER OF CODEBOOKS

| 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% |

recognition accuracy of 64.07 percent. With right-context-dependent phone modeling, 4473 were correctly recognized, increasing the recognition accuracy to 73.80 percent. 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. Our DP algorithm considers substitutions and deletions as errors, and tries to minimize the number of errors. 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 the four broad classes (sonorant, stop, fricative, and closure) are reported in Table III. For these experiments, a bigram phone-language model is used, insertions are *not* counted as errors, and are held to 10–12 percent by appropriately weighing the language model and acoustic model probabilities. These conditions are identical to that used by Schwartz *et al.* [3].

### *B. Additional Experiments*

*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.* [3] 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 subconsciously 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* probabilities of the phones, which is the same as 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 language models, and present our results in Table IV. As expected, the use of simpler language models led to some degradations.

*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 V shows the results for context-independent phone modeling. To use multiple feature sets in a codebook, interset distances are computed and combined using a linear combination [13]. The weights in the linear combination were optimized from earlier experiments using a different database. The weights used for cepstrum, differenced cepstrum, differenced power, and power are: 1, 0.8, 0.01, and 0.05, respectively.

We find that using only one set of the features in one codebook produced poor results. As expected, power gave much worse results than cepstrum or differenced cepstrum coefficients. Linearly combining all three sets of a features in one codebook [22], [13] 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. This illustrates the utility of the additional features, as well as the additional codebooks.

*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 phone models for this experiment. We found that with 2830 training sentences, the results were not significantly different. This is 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 Fig. 4. 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.

### C. Discussion

Without using lexical or higher level knowledge, expert spectrogram readers could recognize phones from continuous speech with an accuracy of 69 percent [11]. 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 Zue [23], 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, and thereby increases recognition accuracy. It was precisely because of this lack of uniformity that we believe 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 phone recognition for several years. Their earlier results can be found in [24]. 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 percent [25].

Nakagawa [26] applied statistical pattern classification and dynamic time warp to speaker-independent phone recognition. He reported 51 and 56 percent with these two approaches for 7 vowels, 71 and 74 percent for 9 fricatives, and 57 and 55 percent 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 [27] used artificial neural networks for the recognition of 16 vowels, and reported 54 percent for context-independent recognition, and 67 percent 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.

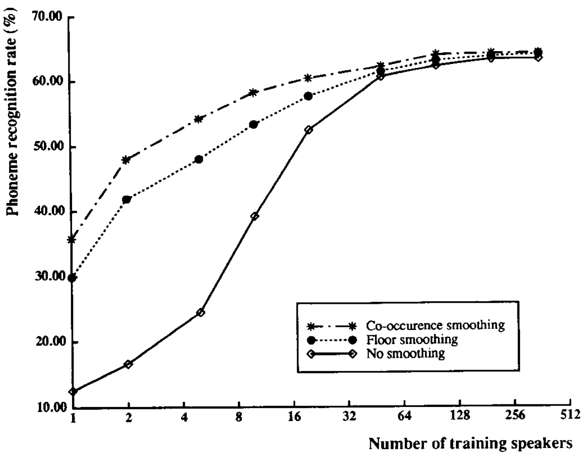

Fig. 4. Phone recognition results with context-independent models, using different number of training speakers and smoothing algorithms.

Another interesting comparison is BBN's speaker-*dependent* phone recognizer [3]. They reported phone recognition rates of 61 and 79 percent for one very good speaker using context-independent and left-context-dependent models, respectively [3]. 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 many more training data that are 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. One possible explanation is our use of differential parameters with context-independent models, which already accounted for some contextual 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 [3], 3) use of continuous parameters [28], [5], 4) use of maximum mutual information estimation [28], 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.

## V. Conclusion

In this paper, we extended the currently popular hidden Markov modeling technique to speaker-independent phone recognizer. 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 percent 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 [15].

## Acknowledgment

The authors wish to thank R. Reddy for his motivation and guidance; K. Shikano for providing the LPC routines; and R. Cole, R. Stern, and V. Zue for reading drafts of this paper.

## References

[1] A. Averbuch *et al.*, "Experiments with the Tangora 20,000 word speech recognizer," presented at the IEEE Int. Conf. Acoust., Speech, Signal Processing, Apr. 1987.

[2] Y. L. Chow, M. O. Dunham, O. A. Kimball, M. A. Krasner, G. F. Kubala, J. Makhoul, S. Roucos, and R. M. Schwartz, "BYBLOS: The BBN continuous speech recognition system," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Apr. 1987, pp. 89-92.

[3] R. Schwartz, Y. Chow, O. Kimball, S. Roucos, M. Krasner, and J. Makhoul, "Context-dependent modeling for acoustic-phonetic recognition of continuous speech," presented at the IEEE Int. Conf. Acoust., Speech, Signal Processing, Apr. 1985.

[4] L. R. Rabiner, B. H. Juang, S. E. Levinson, and M. M. Sondhi, "Recognition of isolated digits using hidden Markov models with continuous mixture densities," *AT&T Tech. J.*, vol. 64, no. 6, pp. 1211-1233, July-Aug. 1985.

[5] L. R. Rabiner, J. G. Wilpon, and F. K. Soong, "High performance connected digit recognition using hidden Markov models," presented at the IEEE Int. Conf. Acoust., Speech, Signal Processing, Apr. 1988.

[6] J.-P. Haton, "Knowledge-based and Expert systems in automatic speech recognition," in *New Systems and Architectures for Automatic Speech Recognition and Synthesis*, R. DeMori, Ed. The Netherlands: Dordrecht, Reidel, 1984.

[7] R. A. Cole, M. Phillips, B. Brennan, and B. Chigier, "The C-MU phonetic classification system," presented at the IEEE Int. Conf. Acoust., Speech, Signal Processing, Apr. 1986.

[8] H. S. Thompson and J. D. Laver, "The alvey speech demonstrator—Architecture, methodology, and progress to date," *Proc. Speech Tech.*, 1987.

[9] L. F. Lamel, R. H. Kassel, and S. Seneff, "Speech database development: Design and analysis of the acoustic-phonetic corpus," in *Proc. DARPA Speech Recogn. Workshop*, L. S. Baumann, Ed., Feb. 1986, pp. 100-109.

[10] W. M. Fisher, V. Zue, J. Bernstein, and D. Pallett, "An acoustic-phonetic data base," presented at the 113th Meet. Acoust. Soc. Amer., May 1987.

[11] R. Weide, personal communication, unpublished.

[12] A. V. Oppenheim and R. W. Schafer, *Digital Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1975.

[13] K. Shikano, "Evaluation of LPC spectral matching measures for phonetic unit recognition," Tech. Rep., Comput. Sci. Dep., Carnegie Mellon Univ., May 1985.

[14] V. N. Gupta, M. Lennig, and P. Mermelstein, "Integration of acoustic information in a large vocabulary word recognizer," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Apr. 1987, pp. 697-700.

[15] K. F. Lee, "Large-vocabulary speaker-independent continuous speech recognition: The SPHINX system," Ph.D. dissertation, Comput. Sci. Dep., Carnegie Mellon Univ., 1988.

[16] K. Shikano, K. Lee, and D. R. Reddy, "Speaker adaptation through vector quantization," presented at the IEEE Int. Conf. Acoust., Speech, Signal Processing, Apr. 1986.

[17] F. Jelinek, "Continuous speech recognition by statistical methods," *Proc. IEEE*, vol. 64, pp. 532-556, Apr. 1976.

[18] R. Schwartz *et al.*, "Towards robust hidden Markov models," presented at the DARPA Workshop Speech Recogn., June 1988.

[19] F. Jelinek and R. L. Mercer, "Interpolated estimation of Markov source parameters from sparse data," in *Pattern Recognition in Practice*, E. S. Gelsema and L. N. Kanal, Eds. Amsterdam, The Netherlands: North-Holland, 1980, pp. 381-397.

[20] K. Sugawara, M. Nishimura, and A. Kuroda, "Speaker adaptation for a hidden Markov model," presented at the IEEE Int. Conf. Acoust., Speech, Signal Processing, Apr. 1986.

[21] A. J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Trans. Inform. Theory*, vol. IT-13, pp. 260-269, Apr. 1967.

[22] S. Furui, "Speaker-independent isolated word recognition using dynamic features of speech spectrum," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, pp. 52-59, Feb. 1986.

[23] R. A. Cole, A. I. Rudnicky, V. W. Zue, and D. R. Reddy, "Speech as patterns on paper," in *Perception and Production of Fluent Speech*, R. A. Cole, Ed. Hillsdale, NJ: Lawrence Erlbaum Associates, 1980.

[24] R. A. Cole, "Phonetic classification in new generation speech recognition systems," *Speech Tech. 86*, pp. 43-46, 1986.

[25] B. Chigier, personal communication, unpublished.

[26] S. Nakagawa, "Speaker-independent phoneme recognition in continuous speech by a statistical method and a stochastic dynamic time warping method," Tech. Rep. CMU-CS-86-102, Comput. Sci. Dep., Carnegie Mellon Univ., Jan. 1986.

[27] H. C. Leung and V. W. Zue, "Some phonetic recognition experiments using artificial neural nets," presented at the IEEE Int. Conf. Acoust., Speech, Signal Processing, Apr. 1988.

[28] P. Brown, "The acoustic-modeling problem in automatic speech recognition," Ph.D. dissertation, Comput. Sci. Dep., Carnegie Mellon Univ., May 1987.

**Kai-Fu Lee** (S'85-M'88) was born in Taipei, Taiwan, in 1961. He received the A.B. degree (summa cum laude) in computer science from Columbia University, New York, NY, in 1983, and the Ph.D. degree in computer science from Carnegie Mellon University, Pittsburgh, PA, in 1988.

Since May 1988 he has been a Research Computer Scientist at Carnegie Mellon, where he currently directs the speech recognition effort within the speech group. His current research interests include automatic speech recognition, spoken language systems, artificial intelligence, and neural networks.

Dr. Lee is a member of Phi Beta Kappa, Sigma Xi, the Acoustical Society of America, and the American Association of Artificial Intelligence.

**Hsiao-wuen Hon** was born on May 31, 1963. He received the B.S. degree in electrical engineering from National Taiwan University in 1985.

Since 1986 he has been a Ph.D. student in the Computer Science Department, Carnegie-Mellon University, Pittsburgh, PA, where he is involved in speech research. From 1985 to 1986 he was a full-time Teaching Assistant in the Department of Computer Science and Information Engineering, National Taiwan University. His research interests include speech recognition, artificial intelligence, neural network, pattern recognition, stochastical modeling, and signal processing.