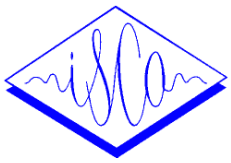

First International Conference on
Spoken Language Processing (ICSLP 90)
Kobe, Japan
November 18-22, 1990

Vocabulary Independent Phrase Recognition With a Linear Phonetic Context Model

Yoshiharu Abe and *Kunio Nakajima*

Information Systems and Electronics
Development Laboratory
Mitsubishi Electric Corporation
5-1-1, Ofuna, Kamakura, 247 Japan

Abstract

This paper describes a continuous speech recognition system using phonemes as basic units in both of acoustic and linguistic processings. Acoustically the phoneme is difficult to be identified since it is strongly varied by the phonemic contexts. In order to cope with the contextual variability of the phonemes, we use a linear model called Linear Phonetic-Context Model(LPCM), which represents acoustical features as the sum of context-independent and context-dependent components. Incorporating with the LPCM we design a phoneme-based phrase recognition algorithm which accepts speech input of an arbitrary string of phonemes. The algorithm obtains plural recognition candidates using an end-point spotting method along with a beam-search technique. The language model bases on task-independent statistics of phoneme strings, and gives a probability of a phrase not existing in the corpus avoiding the null probability problem as in the simple N-gram model. In experiments to recognize 336 phrases extracted from 50 sentences spoken by a male speaker, we obtained a phoneme recognition rate of 95.0% and a phrase recognition rate of 67.9% without limiting a vocabulary.

1. Introduction

In large vocabulary speech recognition it is a natural way to use phonemes as basic units since our language can be transcribed by a small set of phonemes. An important point in this approach is to develop an accurate model for acoustic manifestation of phonemes as acoustic features.

It is well known that the acoustic features of the phonemes are strongly varied by the phonemic contexts because of coarticulation. To cope with this problem we have proposed a linear model for representing the dependence of the acoustic features of phonemes on the acoustic/phonetic contexts[1]. In this contextual model, called the Linear Phonetic-Context Model(LPCM), the acoustic features in a phonemic segment is represented as the sum of context-independent and context-dependent components. The parameters representing the contextual dependence and the context-independent components are algebraically obtained by a maximum-likelihood estimation method.

In the case of isolated word recognition, we have shown that the LPCM was superior than the DP-matching method with whole word templates and the error rate was reduced to one third of the value obtained by the latter method[2,3].

In this paper we describe an experimental system for recognizing continuous speech spoken without limiting a vocabulary. The system is comprised of two components corresponding to acoustic and linguistic processings. In the both components phonemes are used as basic units.

In the acoustic processing the system obtains plural candidates of phoneme strings hypothesizing an arbitrary string of phonemes for the input speech. To avoid computational explosion in the recognition process an end-point spotting method along with a beam-search technique is used.

The linguistic processing of the phoneme strings is performed by a statistical language model based on statistics of subwords corresponding an arbitrary portion of phoneme strings existing in a training corpus. Since the subwords are mechanically extracted, the language model is independent of a task and can be considered to be suitable for Japanese as an agglutinative language.

Finally we test the system through speaker-dependent speech recognition experiments using phrases extracted from continuously spoken sentences.

2. The Linear Phonetic-Context Model

In this section we review the contextual model that has been proposed in the literature[1] to represent the dependence of the acoustic features of phonemes on the acoustic/phonetic contexts. Since the model represents the contextual dependence of the acoustic features by a linear model we call it the linear phonetic-context model(LPCM) hereafter.

2.1 The Context Model for Acoustic Features

Let $\{\boldsymbol{x}_N(i), i = 1,\ldots,I\}$ be N-dimensional feature vectors obtained by analyzing speech signals produced by continuously speaking a consecutive sequence of phonemic symbols $\{q_j, j = 1,\ldots,J\}$ and R be a number of stochastic states used to represent acoustic manifestation of a phoneme. And let us assume we can uniquely determine the state and the phoneme from which an observed feature vector has emerged.

Then the feature vector $\boldsymbol{x}_N(i)$ emerging from the stochastic state $r \in [1,R]$ of the phoneme $q_j \in [1,Q]$ is represented as

$$\boldsymbol{x}_N(i) = \boldsymbol{a}_N(q_j,r) + \boldsymbol{B}_{N,L}(q_j,r)\boldsymbol{z}_L(i,j) + \epsilon_N(i) \qquad (1)$$

where $\boldsymbol{a}_N(q_j,r)$ is an N-dimensional context-independent vector, $\boldsymbol{B}_{N,L}(q_j,r)$ is a contextual dependence matrix with N-rows and L-columns, $\boldsymbol{z}_L(i,j)$ is an L-dimensional context vector obtained from the acoustic/phonetic contexts of the feature vector $\boldsymbol{x}_N(i)$ and $\epsilon_N(i)$ is an estimation error vector with a multivariate normal density with mean 0_N and covariance $\boldsymbol{C}_{N,N}(q_j,r)$.

In our system, the context vector $z_L(i,j)$ is defined by

$$z_L(i,j) = [z_{L_1}(i)^\top, z_{L_2}(j)^\top]^\top \tag{2}$$

$$L = L_1 + L_2 \tag{3}$$

where z_{L_1} and z_{L_2} are vectors representing bottom-up and top-down contexts, respectively.

By using the feature vectors at time $i \pm \delta$, the bottom-up context vector z_{L_1} is given by

$$z_{L_1}(i) = [x_N(i-\delta)^\top, x_N(i+\delta)^\top]^\top \tag{4}$$

$$L_1 = 2N \tag{5}$$

and by using the identity numbers of left and right phonemes, q_{j-1} and q_{j+1}, the top-down context vector is given by

$$z_{L_2}(j) = [e_Q(q_{j-1})^\top, e_Q(q_{j+1})^\top]^\top \tag{6}$$

$$L_2 = 2Q \tag{7}$$

where, $e_Q(q)$ denotes a Q-dimensional unit length vector defined by

$$e_Q(q) = [\overbrace{0\cdots0}^{q-1 \text{zeros}} \; 1\, 0\cdots0] \tag{8}$$

2.2 The Context Model for Duration

We also represent the dependence of duration on the phonemic contexts by a similar model to equation (1).

Let $d^s(j,r)$ be duration of r-th state of the phoneme q_j and $d^p(j)$ be duration of the phoneme q_j. Those durations are modeled as

$$d^s(j,r) = a^s(q_j,r) + b_L^s{}^\top(q_j,r) z_{L_2}(j) + \epsilon^s(j,r) \tag{9}$$

and

$$d^p(j) = a^p(q_j) + b_L^p{}^\top(q_j) z_{L_2}(j) + \epsilon^p(j) \tag{10}$$

Note that as for the durational models we only use a top-down context vector, z_{L_2}.

2.3 Estimation of Parameters

We have to estimate the parameters, $\boldsymbol{a}_N(q,r)$, $\boldsymbol{B}_{N,L}(q,r)$ and $\boldsymbol{C}_{N,N}(q,r)$, for states $r \in [1,R]$ of phonemes $q \in [1,Q]$. For estimation of the parameters for the state r of the phoneme q we use a training set comprised of pairs of feature vectors and context vectors as

$$\Omega^x(q,r) = \{x_N(k), z_L(k) | k = 1, \ldots, K(q,r)\} \tag{11}$$

where $z_L(k)$ is a context vector of a feature vector $x_N(k)$. Those training data can be obtained from a corpus of labeled speech data.

Maximum likelihood estimations of the parameters, $\boldsymbol{a}_N(q,r)$ and $\boldsymbol{B}_{N,L}(q,r)$, are algebraically given by

$$\hat{\mathbf{a}}_N(q,r) = (\mathbf{u}_N - \hat{\boldsymbol{B}}_{N,L}(q,r)\mathbf{u}_L)/K(q,r) \tag{12}$$

$$\begin{aligned}\hat{\boldsymbol{B}}_{N,L}(q,r) = &(\boldsymbol{V}_{N,L} - \mathbf{u}_N\mathbf{u}_L^\top/K(q,r)) \\ &\cdot (\boldsymbol{V}_{L,L} - \mathbf{u}_L\mathbf{u}_L^\top/K(q,r))^+\end{aligned} \tag{13}$$

where,

$$\begin{aligned}\mathbf{u}_N &= \sum_{k=1}^{K(q,r)} \mathbf{x}_N(k), & \boldsymbol{V}_{N,L} &= \sum_{k=1}^{K(q,r)} \mathbf{x}_N(k)\mathbf{z}_L(k)^\top \\ \mathbf{u}_L &= \sum_{k=1}^{K(q,r)} \mathbf{z}_L(k), & \boldsymbol{V}_{L,L} &= \sum_{k=1}^{K(q,r)} \mathbf{z}_L(k)\mathbf{z}_L(k)^\top\end{aligned} \tag{14}$$

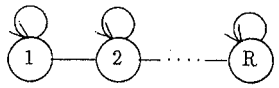

Figure 1 A Phoneme Model.

1	a	[a]	13	h	[h] [ç] [f]
2	i	[i] [i̥]	14	b	[b]
3	u	[ɯ] [ɯ̥]	15	d	[d]
4	e	[e]	16	g	[g] [ŋ]
5	o	[o]	17	z	[z] [dz]
6	j	[j]	18	m	[m]
7	w	[w]	19	n	[n]
8	p	[p]	20	r	[r]
9	t	[t]	21	Q	closure before /p,t,
10	k	[k]			k,c,s,h/('soku-on')
11	c	[ts] [tʃ]	22	N	[m] [n]
12	s	[s] [ʃ]	23	#	(silence)

Figure 2 The Phonemic Units with Their Phonetic Variations.

and $\cdot^+$ denotes generalized inversion of a matrix.

Using the estimations, $\hat{\mathbf{a}}_N(q,r)$ and $\hat{\boldsymbol{B}}_{N,L}(q,r)$, the covariance matrices are given by

$$\hat{\boldsymbol{C}}_{N,N}(q,r) = \frac{1}{K(q,r)} \sum_{k=1}^{K(q,r)} \hat{\epsilon}_N(k)\hat{\epsilon}_N(k)^\top \tag{15}$$

where

$$\hat{\epsilon}_N(k) = x_N(k) - \hat{\mathbf{a}}_N(q,r) - \hat{\boldsymbol{B}}_{N,L}(q,r) z_L(k) \tag{16}$$

2.4 Parameter Reestimation

Since the boundaries in the speech database are not necessarily optimum for the LPCM, we reestimate the boundaries using an iterative procedure. The procedure performs two steps at each iteration.

The first step at the i-th iteration is to estimate a set of the parameters of the LPCM, $\mathcal{P}^{(i)}$, based on a set of the current boundaries, $\mathcal{B}^{(i)}$. The second step is to estimate the next set of the boundaries, $\mathcal{B}^{(i+1)}$, based on the current set of parameters, $\mathcal{P}^{(i)}$. The above two steps are repeated until the some convergence criterion is reached.

3. The Recognition Algorithm

The task of the recognition algorithm is to find plural candidates of phoneme strings for the input speech without using any linguistic knowledge, or to merely find more probable phonemic transcriptions for the input speech.

For this purpose, we implement the algorithm based on a hypothesis-and-test scheme. In the hypothesis phase the algorithm considers an arbitrary string of phonemes by expanding an old phoneme string with appending an arbitrary phoneme to it. To avoid computational explosion in the recognition process we employ a beam-search technique and an end-point spotting method.

In the test phase the algorithm generates the model of speech corresponding to the phoneme string based on the LPCM and calculates a score of the generated model to the input speech using a DP-matching method. Pruning the hypotheses only at the spotted end-points and saving the intermediate results of the DP-matching for the next level, the algorithm becomes farely efficient. Although the obtained results does not necessarily coincide with the full-search results, we can obtain almost optimum results.

The functional block of the algorithm is shown in Figure 3. The hypothesis table holds the hypotheses generated in the recognition process. The hypothesis is defined as a 6-tuple

$$\mathbf{H} =< W, E, S, G^x, G^d, M >$$

where W, E, S, denote a phoneme string, a spotted end-point and a score of the hypothesis, respectively, G^x and G^d denote storages to save accumulated likelihoods of the feature vectors and durations at a level of the DP-matching and M is a mark to show that the hypothesis has been expanded.

With those definition the algorithm is given as follows:

(i) Clear the hypothesis table and put a hypothesis, with an empty phoneme string λ,

$$\mathbf{H}_0 =< \lambda, 1, \infty, G^x, G^d, \text{false} >$$

where

$$G^x(0) = 0, \quad G^x(i) = \infty (i = 1, 2, \ldots)$$
$$G^d(0) = 0, \quad G^d(i) = \infty (i = 1, 2, \ldots)$$

For each time $i = 1, 2, \ldots,$ execute the following steps:

(ii) Select hypotheses with $E = i$ and $M =$ **false** from the hypothesis table.

(iii) Compare the scores of the selected hypotheses and remain top B hypotheses, $\{\mathbf{H}_k, k = 1, \ldots, B\}$, for the next step.

For each index k, execute the following steps.

(iv) Expand the hypothesis, $\mathbf{H}_k$, by appending an arbitrary phoneme $q_{j_k} \in [1, Q]$, where j_k denotes the length of the phoneme string of the hypothesis $\mathbf{H}_k$.

(v) 1. Generate the models, based on the LPCM, for the phoneme string, $q_1, \ldots, q_{j_k}$, assuming that an arbitrary phoneme, $q_{j_k+1} \in [1, Q]$, exists on the right of the phoneme, q_{j_k},

2. match those models with the input speech using a DP-matching algorithm and

3. spot an end point that gives a maximum score.

In the generation of the model, we have to pay attention to the range of the influence by the alternation of the phonemic contexts. In our case the range is limited to one phonemic unit (see equation (6)).

(vi) 1. Pool the hypothesis expanded in the hypothesis table for further processings along with the accumulated likelihoods obtained in the DP-matching and

2. set the mark of the seed hypothesis $\mathbf{H}_k$ with **true** to inhibit this hypothesis is again used as the seed in step (iv).

The score of the hypothesis consists of likelihoods of the feature vector, durations and language. To normalize different contributions of each component to the score, we calculate the score as

$$S = \frac{1}{i}\sum_{t=1}^{i} l^x(t) + \frac{\alpha^s}{\gamma j}\sum_{k=1}^{jR} l^{d^s}(k) + \frac{\alpha^p}{\gamma j}\sum_{k=1}^{j} l^{d^p}(k) + \frac{\beta}{j} l^w(q_1 \cdots q_j)$$

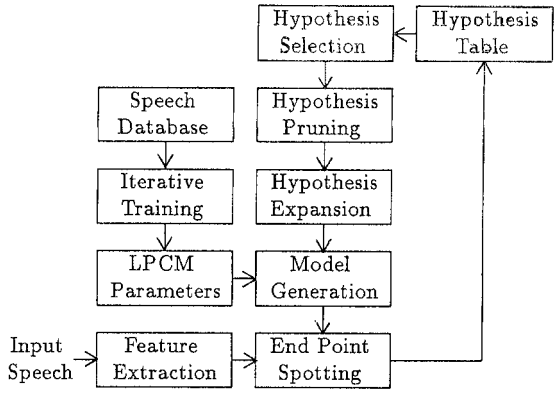

Figure 3 A Functional Block Diagram of the Speech Recognition System with an Unlimited Vocabulary.

where $l^x(t)$, $l^{d^s}(k)$, $l^{d^p}(k)$ and $l^w(W)$ denote likelihoods of the feature vector, durations for state and phoneme, and language, respectively, α^s, α^p and β are weights for respective terms, and γ is duration of phoneme averaged over all phonemes. In the following section we describe a language model for estimating the likelihood of language.

4. The Language Model for Phoneme Strings

The statistical language modeling is used to realize a task-independent linguistic processing. We count the occurrences of an arbitrary string of phonemic symbols in a large text database limiting the length of the string by L according to the size of storage allowed. These statistics of subword-length strings are used to estimate a probability of a given phoneme string w.

The N-gram model based on the markov model of the order, $N-1$, has a problem that if we increased the number N to strengthen the constraint by language, the zero probability occur. To avoid the problem, we divide the phoneme string into subwords that are frequently observed in the training corpus and obtain the probability of the phoneme string by maximizing the product of probabilities of the subwords and the transition probabilities among the subwords. The maximization with an optimum division can be obtained automatically by a dynamic programming algorithm.

Let $\Pr(w)$ be a probability of the phoneme string w by the language model and q_j be the j-th phonemic symbol in the phoneme string w.

The probability $\Pr(w)$ is given by

$$\Pr(w) = \max_{x, w_1, w_2, \ldots, w_x} \prod_{i=1}^{x} \Pr(w_i) \prod_{i=1}^{x-1} \Pr(w_{i+1} | w_1 \cdots w_i) \qquad (17)$$

where w_i denotes the i-th subword satisfying

$$w = w_1 w_2 \cdots w_x \qquad (18)$$

and x is the number of the divisions. We approximate the transition probability, $\Pr(w_{i+1}|w_1 \cdots w_i)$, as

$$\begin{aligned} &\Pr\{w_{i+1}|w_1 \cdots w_i) \\ &\approx \Pr\{\text{last}(M-1, w_1 \cdots w_i)|\text{first}(1, w_{i+1})\} \end{aligned} \qquad (19)$$

where $\mathrm{last}(N, w)$ and $\mathrm{first}(N, w)$ denote the last and the first N phonemic symbols of the string w.

The maximization in equation (17) is done by solving a DP-equation

$$g(j) = \max_{0 \le t \le j-1} g(t) \Pr(q_{t+1} \cdots q_j) \Pr\{\mathrm{first}(1, q_{t+1} \cdots q_j) | \mathrm{last}(M-1, q_1 \cdots q_t)\} \quad (20)$$

with an initial condition

$$g(0) = 1 \quad (21)$$

Now we get the probability $\Pr(w)$ as $g(J)$, where J is the length of the phoneme string w.

5. Experimental Results

In order to evaluate the performance of the system we conducted speaker-dependent recognition of phrasal speech with an unlimited vocabulary.

5.1 Results with No Language Models

Firstly we have evaluated the system without using the language model. The test set consists of 336 phrases extracted from 50 sentences continuously spoken by a male narrator. For the training of the LPCM we used a phonemically balanced corpus of 400 sentences spoken by the same speaker. The δ in equation (4) for the bottom-up contexts is set with 5. The speech data was sampled at 10 kHz and was analyzed by the LPC with order of 15 at a 100 Hz frame rate. The first 10 mel-cepstral coefficients excluding the energy term were used as the acoustic features. The beam width is set to be 50.

The results are shown in **Table I**. 10.7% of the test phrases did not remain in the beam. The phrase accuracy was calculated including those drops. The phoneme accuracy was calculated from the phrase recognition results considering the numbers of substitutions, insertions and deletions. The test phrases have 7.14 phonemes on the average.

Obviously we obtained a high phoneme accuracy of 91.4% with very small deletion errors as compared to the substitution and insertion errors. This shows that the effectiveness of the LPCM to the continuous speech. The phrase accuracy, 46.1%, approximately equals to a value, 52.6%, obtained by raising the phoneme accuracy, 91.4%, to the power of the average length of the test phrases, 7.14 (i.e., $0.526 \approx 0.914^{7.14}$.)

5.2 Results with Language Models

Secondly we have evaluated the language model. The statistics of the language model were obtained from a text corpus maximumly comprised of $191K$-phrases which are transcribed by $1489K$-phonemic symbols. The content of the text corpus has no relation to the contents of the spoken materials described above.

The results are shown in **Table II**. For the sake of comparison, the results with the phonemic tri-gram model are also shown.

The proposed language model(called 'connected subword model') obtained maximum phoneme accuracy of 95.0% and phrase accuracy of 67.9%, when the maximum subword length L was 16 and the size of the training corpus was large. Increasement of the size of the training corpus raises the accuracies more than the increasement of the maximum length of subwords, L. Those results encourage us to collect more texts for the training corpus.

In above experiment we fixed the M in equation (19) to 1, which means we only used probabilities of the first phonemes of the subwords as the transition probabilities. For other values of M, the results are shown in **Table III**. But increasing value of M did not improve the accuracies.

Table I Experimental Results with No Language Models.

Phrase Accuracy	Phoneme Accuracy	Percent Insertion	Percent Deletion	Percent Substitution
46.1%	91.4%	4.8%	0.6%	3.2%

Table II The Results with Language Models, for Various Values of L (M=1).

Training Corpus	Accuracy Unit	Connected Subword Model				Trigram Model
		L=10	L=12	L=14	L=16	
370Kpho.	Phoneme	93.8	94.0	94.0	94.0	92.7
48Kphr.	Phrase	61.6	62.5	62.8	62.8	55.4
1086Kpho.	Phoneme	94.1	94.2	94.2	94.2	92.1
133Kphr.	Phrase	63.4	63.7	64.0	64.0	53.6
1489Kpho.	Phoneme	94.9	95.0	95.0	95.0	92.3
191Kphr.	Phrase	67.3	67.6	67.9	67.9	53.0

Table III The Results with Language Models, for Various Values of M.

Training Corpus	Accuracy Unit	L=9			L=16		
		M=1	M=2	M=3	M=1	M=2	M=3
370Kpho.	Phoneme	93.9	93.7	93.7	94.0	93.8	93.8
48Kphr.	Phrase	61.6	61.3	61.9	62.8	62.2	62.5
1086Kpho.	Phoneme	94.0	93.5	93.9	94.2	93.6	94.0
133Kphr.	Phrase	62.8	62.8	63.4	64.0	63.7	64.0
1489Kpho.	Phoneme	94.7	94.5	94.7	95.0	94.6	94.8
191Kphr.	Phrase	65.8	65.8	66.7	67.9	66.4	67.3

6. Conclusion

We have described a continuous speech recognition system using the Linear Phonetic-Context Model in the acoustic processing. The system uses the phonemes as the basic units in both of the acoustic and linguistic processings, and does not limit a usable vocabulary. The recognition algorithm was designed to allow speech input of an arbitrary phoneme string. To avoid computational explosion we employed an end-point spotting method along with a beam-search technique. By using the language model based on the statistics of subwords existing in a large text corpus, we have realized a task-independent linguistic processing. Finally we tested the system using phrasal speech and obtained good results encouraging us for further investigation.

References

[1] Abe, Y. and Nakajima, K., "Speech Recognition Using Dynamic Transformation of Phoneme Templates Depending on Acoustic/Phonetic Environments," Proc. IEEE Int. Conf. on Acoust. Speech and Signal Processing, London, May, 1989, pp326–329.

[2] Abe,Y. and Nakajima, K., "Speech Recognition Method Using a Linear Phonetic-Context Model," Trans. Inst. Electron. Inf. Commun. Eng. Japan, J72-D-II, 8, pp.1228-1233, 1989.

[3] Abe,Y. and Nakajima, K., "Large Vocabulary Word Recognition Using a Linear Context Model," Proc. Spring Meeting of Acoust. Society of Japan, 1-6-7, 1989.