論文 / 著書情報
Article / Book Information

論題(和文)	
Title	MDL-based context-dependent subword modeling for speech recognition
著者(和文)	篠田 浩一
Author	K. Shinoda, T. Watanabe
出典(和文)	日本音響学会英文論文誌, Vol. 21, No. 2, pp. 79-86
Journal/Book name	Journal of Acoustic Society of Japan (E), Vol. 21, No. 2, pp. 79-86
発行日 / Issue date	2000,

J. Acoust. Soc. Jpn. (*E*) **21**, 2 (2000)

PAPER

MDL-based context-dependent subword modeling for speech recognition

Koichi Shinoda and Takao Watanabe

NEC Corporation,
4-1-1, Miyazaki, Miyamae-ku, Kawasaki, 216-8555 Japan

(*Received 17 March 1999*)

Context-dependent phone units, such as triphones, have recently come to be used to model subword units in speech recognition systems that are based on the use of hidden Markov models (HMMs). While most such systems employ clustering of the HMM parameters (*e.g.*, subword clustering and state clustering) to control the HMM size, so as to avoid poor recognition accuracy due to a lack of training data, none of them provide any effective criteria for determining the optimal number of clusters. This paper proposes a method in which state clustering is accomplished by way of phonetic decision trees and in which the minimum description length (MDL) criterion is used to optimize the number of clusters. Large-vocabulary Japanese-language recognition experiments show that this method achieves higher accuracy than the maximum-likelihood approach.

Keywords: Speech recognition, Acoustic modeling, Context-dependent phone, State clustering, MDL criterion

PACS number: 43. 72. Ne

1. INTRODUCTION

Over the past few years, extensive studies have been carried out on speaker-independent speech recognition systems that employ continuous density hidden Markov models (HMMs). It is well known that in most such systems the use of context-dependent (CD) phone models rather than context-independent (CI) phone models (monophones) provides greater recognition accuracy.[1-10].

While the large number of CD models employed in a typical system can help to capture variations in speech data, the amount of available training data is likely to be insufficient to support the use of such a large number. Furthermore, there is great variation in the frequency with which individual CD phone units can be expected to appear in training data; in most CD phone unit sets, the frequencies for some units will be so small that they will be unlikely to appear in training data even when a very large amount of data is provided. Such lack of data can seriously degrade speech recognition performance and most recognition systems using CD models cluster the model parameters to try to alleviate the problem.

Various clustering methods have been developed for this purpose. One variation among them is the choice of parameter to be clustered: K.-F. Lee *et al.*,[1] for example, use subword clustering, Hwang *et al.*[2] use state clustering, and Digalakis *et al.*[3] cluster the mixture components of the HMMs with Gaussian-mixture state observation densities. There is also variation in the approach to selecting the acoustically-similar parameters to be clustered. One approach is to use only the acoustic characteristics of the data.[2-6] Another approach is to utilize *a priori* knowledge about acoustic similarities (usually represented in the form of decision trees) between the parameters, in addition to the acoustic characteristics themselves.[1,7-9]

However clustering is performed, the accuracy with which the acoustic similarities are measured

will be extremely important. One of the most successful approaches in this regard is that based on the maximum-likelihood (ML) criterion (*e.g.*, Ref. 9)). In this approach, a calculation is made for each parameter cluster in the model to determine the degree to which the splitting of that cluster would increase the likelihood of the model's outputting the training data; the cluster giving the greatest increase is then split. (Here, for the sake of simplicity, we are considering only the "splitting" method (*i.e.*, top-down clustering), but an explanation of the application of ML to bottom-up clustering would be quite similar.)

The difficulty with the ML approach, however, is determining when to halt the splitting process, which could be carried on until the model simply consisted of a full set of individual, unclustered parameters. Most methods limit splitting by imposing a threshold value on the increase in the likelihood or on the number of parameter clusters, but the process required to optimize such thresholds (a series of recognition experiments; cross-validation; *etc.*) is computationally expensive.

In this paper we propose a new approach that uses the minimum description length (MDL) criterion[12] for state splitting.[11] This MDL approach is effective for deciding when to stop splitting.

This paper is organized as follows: Section 2 briefly reviews the MDL criterion; Section 3 outlines state splitting using a phonetic decision tree; Sections 4 and 5 explain in detail how the MDL criterion is applied to state splitting; Section 6 describes the results of an experimental evaluation of our proposed method of state splitting. Finally, Section 7 discusses several issues related to the proposed method.

2. MDL CRITERION

The MDL criterion[12] has been proven to be effective in selecting the optimal model from among various probabilistic models. It selects the model with the minimum description length for given data. When a set of models $\{1,...,i,...,I\}$ is given, the description length $l_i(x^N)$ for data $\{x^N=x_1,...,x_N\}$ and an underlying model i is given by

$$l_i(x^N)=-\log P_{\hat{\theta}^{(i)}}(x^N)+\frac{\alpha_i}{2}\log N+\log I \quad (1)$$

where α_i is the dimensionality (the number of free parameters) of model i and $\hat{\theta}^{(i)}$ represents the maximum likelihood estimates for the parameters $\theta^{(i)}=(\theta_1^{(i)},..., \theta_{\alpha_i}^{(i)})$ of model i. The first term on the right-hand side of (1) represents the code length for data x^N when model i is used as a probabilistic model. This term is identical to the negative of the log likelihood used in the ML criterion. The second term is related to the complexity of model i and the number of data samples, N. The third term is the code length required for choosing model i and is assumed here to be a constant. As a model becomes more complex, the value of the first term decreases and that of the second term increases. The second term works as a penalty imposed for employing a large model size (see Fig. 1). In a comparison among models, the model with the shortest description length l may be considered the one having the most appropriate size and complexity. As may be seen in (1), the MDL criterion does not need any externally given parameters; the optimal model for the data is automatically obtained once a set of models has been specified.

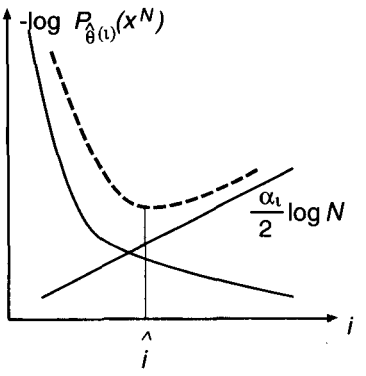

Fig. 1 The MDL criterion.

With complex models of the type used in speech recognition, it is often impractical to calculate the description length for all the possible models because to do so would involve high computational costs. To avoid this, we introduce a number of reasonable assumptions, which are explained in Section 4.

3. TREE-BASED STATE CLUSTERING

In this section, we briefly outline our proposed method. For modeling CD phone units, we use triphones,[10] in which a central phone has left- and right-hand neighbors. Each triphone model is a left-to-right HMM in which states are placed in a line from the start state to the end state, and the transitions with respect to a state consist of that to

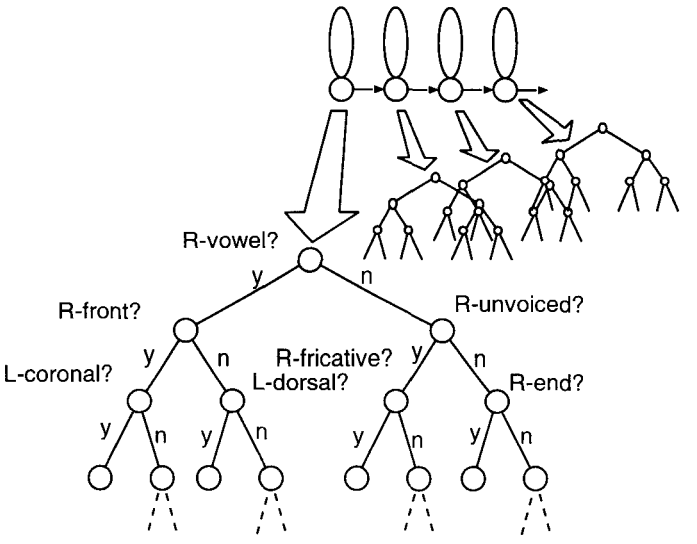

Fig. 2 Phonetic decision tree.

itself and that to the next state to the right. The output density function for each state is a Gaussian probability density function (pdf) for which a diagonal covariance is assumed. All HMMs of triphones whose central phones are the same are assumed to have the same number of states.

As a clustering scheme, we use state splitting based on phonetic decision trees.[9] Those states at the same position in triphone HMMs having the same central phone are pooled into one set, and one phonetic decision tree is constructed for each set. (see Fig. 2.) Starting from the root node which represents the whole set, each node from top to bottom splits off into two other nodes representing, respectively, "yes" or "no" answers to such phonetic-context related questions as: "Is the previous phone unvoiced ?" (**L-unvoiced** ?) and "Is the next phone a fricative ?" (**R-fricative** ?). The MDL criterion is used to choose the optimal question to be asked at each node and to decide when to stop splitting. When all splitting has stopped, the pdf parameters of each leaf node are copied to the pdf parameters of the triphone states in the corresponding subset and used for recognition.

4. DESCRIPTION LENGTH FOR HMMs

4.1 Definition of a Model Set

As explained in Section II, the MDL criterion is used to select an optimal model from among a set of various models. Thus, it is first necessary to prepare the model set from which that optimal model is to be selected. For speech recognition using CDHMMs, it is impossible to prepare all the possible models because of the large number of possible structures of CDHMMs; the number of subword units and/or the number of states in each unit, for example, differ among recognition systems. In this study, we focus on the clustering of the states in CDHMMs and give constant values to those parameters unrelated to state clustering, such as the number of states in a single unit.

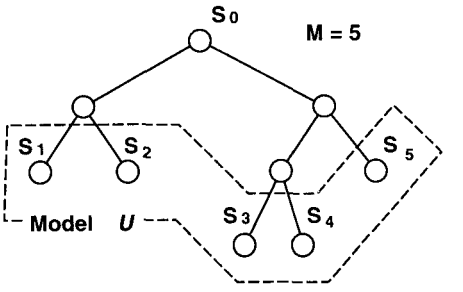

Fig. 3 Model (node set) in the decision tree.

Here a *model* is defined as a node set in a phonetic decision tree in which a Gaussian pdf is assigned for each node. When the root node S_0, which represents the whole set of the triphone states in the tree, is split into M nodes, $S_1,\ldots,S_M$, as shown in Fig. 3, one model $U(S_1,\ldots,S_M)$ is defined for the node set

$\{S_1,\ldots,S_M\}$. Different node sets correspond to different models. The description length for each node set is calculated and the node set with the minimum description length is selected from among various node sets as being the optimum model.

4.2 Calculation of Description Length

Before clustering is performed, an estimate of each HMM parameter is calculated by using the Forward-Backward algorithm.[13] Let speech data for training consist of E examples and each example e be analyzed and represented by a time series of feature vectors, $\{\boldsymbol{o}_1^e,\ldots,\boldsymbol{o}_t^e,\ldots,\boldsymbol{o}_{Te}^e\}$, where T_e is the number of data frames for example e. ML estimates for the Gaussian distribution at state s_l can then be written as:

$$\mu_l=\frac{\sum_{e=1}^{E}\sum_{t=1}^{Te}\gamma_l(e,\ t)\boldsymbol{o}_t^e}{\sum_{e=1}^{E}\sum_{t=1}^{Te}\gamma_l(e,\ t)}, \tag{2}$$

$$\Sigma_l=\frac{\sum_{e=1}^{E}\sum_{t=1}^{Te}\gamma_l(e,\ t)(\boldsymbol{o}_t^e-\mu_l)(\boldsymbol{o}_t^e-\mu_l)'}{\sum_{e=1}^{E}\sum_{t=1}^{Te}\gamma_l(e,\ t)}, \tag{3}$$

where μ_l is the mean vector and Σ_l is the covariance of the Gaussian distribution at state s_l, $(\boldsymbol{o}_t-\mu_l)'$ is the transpose of $(\boldsymbol{o}_t-\mu_l)$, and $\gamma_l(e,\ t)$ is the *a posteriori* probability of the data being in state s_l at the t-th frame of example e, which is calculated as follows:

$$\gamma_l(e,\ t)=\frac{\alpha_l(e,\ t)\beta_l(e,\ t)}{\sum_{l=1}^{L}\alpha_l(e,\ t)\beta_l(e,\ t)}, \tag{4}$$

where L is the total number of all the triphone states in the HMMs, $\alpha_l(e,\ t)$ is the forward probability, and $\beta_l(e,\ t)$ is the backward probability of state s_l at the t-th frame of example e.

The first term on the right-hand side of Eq. (1) is the negative of the log-likelihood of a probabilistic model with respect to given data. It is possible to calculate the log-likelihood of the training data for all the possible node sets, but to do so involves huge computational costs. To reduce these costs, we make the following three assumptions[9]:

1. The transition probabilities of HMMs can be ignored in the calculation of the log-likelihood for a node set.
2. State splitting does not change the frame/state alignment between the data and the model.
3. The log-likelihood of generating the data for each state is the sum of the log-likelihoods of generating each data frame, with each log-likelihood being weighted by the *a posteriori* probability of the data being in the state.

The third assumption is fully justified when the Viterbi algorithm is used for parameter estimation because in this algorithm the posterior probability is either one or zero.

Let us next consider the problem of calculating the description length for the node set U defined in the previous subsection. All the triphone states pooled into the set corresponding the root node S_0, are renumbered as $\{s_1^1,\ldots,s_{L_1}^1,\ s_1^m,\ldots,s_{Lm}^m,\ s_1^M,\ldots,s_{LM}^M\}$, where $\{s_1^m,\ldots,s_l^m,\ldots,s_{Lm}^m\}$ is the subset of states that are merged into node S_m, and L_m is the number of states in the subset. Using Eqs. (2) and (3), the mean vector and the covariance of state s_l^m are written as:

$$\mu_l^m=\frac{\sum_{e=1}^{E}\sum_{t=1}^{Te}\gamma_l^m(e,\ t)\boldsymbol{o}_t^e}{\Gamma_l^m} \tag{5}$$

$$\Sigma_l^m=\frac{\sum_{e=1}^{E}\sum_{t=1}^{Te}\gamma_l^m(e,\ t)(\boldsymbol{o}_t^e-\mu_l^m)(\boldsymbol{o}_t^e-\mu_l^m)'}{\Gamma_l^m}, \tag{6}$$

$$\Gamma_l^m=\sum_{e=1}^{E}\sum_{t=1}^{Te}\gamma_l^m(e,\ t), \tag{7}$$

where $\gamma_l^m(e,\ t)$ is the *a posteriori* probability of the data being in state s_l^m at the t-th frame of example e. Then, under the first and second assumptions, the ML estimates for the pdf parameters of node S_m are given by[14]

$$\mu_m=\frac{\sum_{l=1}^{Lm}\sum_{e=1}^{E}\sum_{t=1}^{T}\gamma_l^m(e,\ t)\boldsymbol{o}_t^e}{\sum_{l}^{Lm}\sum_{e=1}^{E}\sum_{t=1}^{T}\gamma_l^m(e,\ t)}=\frac{\sum_{l=1}^{Lm}\Gamma_l^m\mu_l^m}{\sum_{l}^{Lm}\Gamma_l^m}, \tag{8}$$

$$\Sigma_m=\frac{\sum_{l=1}^{Lm}\sum_{e=1}^{E}\sum_{t=1}^{T}\gamma_l^m(e,\ t)(\boldsymbol{o}_t^e-\mu_l^m)(\boldsymbol{o}_t^e-\mu_l^m)'}{\sum_{l=1}^{Lm}\sum_{e=1}^{E}\sum_{t=1}^{T}\gamma_l^m(e,\ t)}=\frac{\sum_{l=1}^{Lm}\Gamma_l^m(\Sigma_l^m+(\mu_l^m)(\mu_l^m)')}{\sum_{l=1}^{Lm}\Gamma_l^m}-(\mu_m)(\mu_m)', \tag{9}$$

$$\gamma_m(e,\ t)=\sum_{l=1}^{Lm}\gamma_l^m(e,\ t), \tag{10}$$

where μ_m is the mean vector and Σ_m is the covariance of the Gaussian distribution at node S_m. Then, from the third assumption, the approximated log-likelihood L of node S_m generating data $\{\boldsymbol{o}_1,\ldots,\boldsymbol{o}_T\}$ is given by

$$\begin{aligned}L(S_m)&\approx\sum_{e=1}^{E}\sum_{t=1}^{Te}\gamma_m(e,\ t)\log\Bigg(\frac{1}{\sqrt{(2\pi)^K|\Sigma_m|}}\\&\quad\cdot e^{-\frac{1}{2}(\boldsymbol{o}_t^e-\mu_m)'\Sigma_m^{-1}(\boldsymbol{o}_t^e-\mu_m)}\Bigg)\\&=-\sum_{e=1}^{E}\sum_{t=1}^{Te}\frac{1}{2}\gamma_m(e,\ t)(K\log(2\pi)+\log|\Sigma_m|\\&\quad+(\boldsymbol{o}_t^e-\mu_m)'\Sigma_m^{-1}(\boldsymbol{o}_t^e-\mu_m))\\&=-\frac{1}{2}\Gamma_m(K+K\log(2\pi)+\log|\Sigma_m|),\end{aligned} \tag{11}$$

$$\Gamma_m=\sum_{e=1}^{E}\sum_{t=1}^{Te}\gamma_m(e,\ t), \tag{12}$$

where K is the dimensionality of the data vector o_t^e, and Γ_m is the total state occupancy count at node S_m, which is the sum of $\gamma_m(e, t)$ over all data frames of all the examples. The log-likelihood of the data for all the nodes in set U is:

$$L_{all}=\sum_{m=1}^{M} L(S_m)$$
$$=-\sum_{m=1}^{M}\frac{1}{2}\Gamma_m(K+K\log(2\pi)+\log|\Sigma_m|). \quad (14)$$

The second term on the right-hand side of Eq. (1) represents the complexity of a model. In our approach, it is assumed that the covariance of each Gaussian pdf is diagonal. The number of parameters to be estimated for model U is $2KM$ (with model U containing M mean vectors and M diagonal covariances). The total number of data samples is the sum of $\Gamma(S_m)$ over m. With this total, we may approximate the second term as:

$$R=KM\log W, \quad (15)$$

where $W=\sum_{m=1}^{M}\Gamma_m$. As has been previously noted, the third term on the right-hand side of (1) is fixed at a constant value, C, for all possible models.

Finally, using (14) and (15), we may calculate a description length for model U as follows:

$$I(U)=-L_{all}+R+C$$
$$=\sum_{m=1}^{M}\frac{1}{2}\Gamma_m(K+K\log(2\pi)+\log|\Sigma_m|)$$
$$+KM\log W+C. \quad (16)$$

5. STATE SPLITTING USING THE MDL CRITERION

In order to get an optimal model, we need to calculate description lengths for all possible models, which would involve prohibitively high computational costs. Instead, we use an algorithm that obtains only a suboptimal solution.

Let us first assume that node S_m of model U splits into two nodes S_{mqy} and S_{mqn}, in response to question q, and then let $\Delta_m(q)$ be the difference between the description lengths after the splitting and before it (*i.e.*, $l(U')-l(U)$). The description length of model U' is:

$$I(U')=\sum_{m'=1,m'\neq m}^{M}\frac{1}{2}\Gamma_{m'}(K+K\log(2\pi)+\log|\Sigma_{m'}|)$$
$$+\frac{1}{2}\Gamma_{mqy}(K+K\log(2\pi)+\log|\Sigma_{mqy}|)$$
$$+\frac{1}{2}\Gamma_{mqy}(K+K\log(2\pi)+\log|\Sigma_{mqn}|)$$
$$+K(M+1)\log W+C, \quad (17)$$

where the number of nodes for U' is $M+1$, Γ_{mqy} is the state occupancy count for node S_{mqy}, and Γ_{mqn} is that for node S_{mqn}. The difference $\Delta_m(q)$ will then be given by the following equation:

$$\Delta_m(q)=l(U')-l(U)$$
$$=\frac{1}{2}(\Gamma_{mqy}\log|\Sigma_{mqy}|+\Gamma_{mqn}\log|\Sigma_{mqn}|$$
$$-\Gamma_m\log|\Sigma_m|)+K\log W. \quad (18)$$

In state splitting, we first determine the question q' which would minimize $\Delta_0(q')$ when used to split root node S_0. If $\Delta_0(q')>0$, then no splitting is conducted. If $\Delta_0(q')<0$, then node S_0 is split into two nodes, $S_{q'y}$ and $S_{q'n}$, and the same procedure is repeated for each of these two nodes. This node splitting is carried out until there remain no nodes to be split and is conducted for the root nodes of all the phonetic decision trees in all the HMMs.

For the purpose of comparison, let us also consider here the ML approach.[9] Letting $\delta_m(q)$ be the increase in the log-likelihood when node S_m is split into two in response to using question q,

$$\delta_m(q)=L(S_{mqn})+L(S_{mqn})-L(S)$$
$$=-\frac{1}{2}(\Gamma_{mqn}\log|\Sigma_{mqy}|+\Gamma_{mqn}\log|\Sigma_{mqn}|$$
$$-\Gamma_m\log|\Sigma_m|). \quad (19)$$

In the ML approach, question q' which would maximize $\delta_0(q')$ is first chosen from among all the questions, and then it is used to split root node S_0 into two nodes $S_{0q'y}$ and $S_{0q'n}$. This splitting process will continue until stopped by some externally given parameters used to control the number of clusters, since the increase δ is positive in all the splitting. Most methods apply a threshold value to the total occupancy count Γ_m and/or to the log-likelihood increase $\delta_m(q)$. However, the optimization of these parameters requires a series of recognition experiments which are computationally expensive and require additional data. The MDL approach needs no external control parameters; the term $K\log W$ in (18) corresponds to the threshold for increase δ in (19), and this term is estimated automatically on the basis of the training data.

6. EXPERIMENTS

We evaluated our method in experiments testing the recognition of 5,000 Japanese words. Each utterance was digitized at a sampling rate of 16 kHz, and analyzed in 10-ms frame periods. The analysis

yielded a vector of 21 components (a power derivative, 10 mel-scaled cepstral coefficients, and 10 corresponding mel-scaled cepstral time derivatives). We used 37 Japanese CI phones, from which 4,309 triphones were derived. We set the number of states in each HMM to four. A Gaussian output pdf with a diagonal covariance was assumed for each state. The number of questions used in the node splitting was 106. Two data sets, Data A and Data B, were prepared for training. Data A consisted of 250 phonetically-balanced words uttered by each of 46 male speakers. Data B consisted of 2,150 phonetically-balanced words uttered by each of 36 male speakers. Speech data from five other male speakers, none of whom was involved in the production of Data A or Data B, was used for the evaluation tests. Each of these test speakers uttered 250 words. None of the words in the test vocabulary were used in the training vocabulary.

Table 1 shows recognition results obtained with the proposed MDL method (averaged for the five test speakers) and those obtained with the ML approach when Data A was used for training. The various results for the ML approach reflect the different values used for two thresholds, D and V, for the state occupancy count and for the increase in likelihood, respectively.

Let us consider a specific instance in the ML approach. For a node, S, the algorithm determines the set of questions for which neither of the resulting two response nodes would have a "total occupation count" less than or equal to D, and for which $\delta(q)$ would be larger than V. It determines the question among this set for which δ is largest and used it to split node S.

Table 1 shows the results of the ML approach for 17 combinations of D and V(**ML 1-17**). As shown in Table 1, the proposed method achieved higher recognition accuracy than any using the ML approach. In each instance, the computational cost required for the proposed method was roughly the same as that with the ML approach. In order to determine a model of a size optimal for the amount of training data, however, the ML approach must be performed repeatedly over a range of parameter values. Therefore, the total computational cost is much less with the MDL approach proposed here.

Table 2 shows the frequency of the questions used, summed over all the phonetic decision trees of all the HMMs, when the MDL approach was employed. "L-begin" corresponds to the question, "Is the phone located at the beginning of the word ?". Questions related to left phones were used more often than those related to right phones. For a consonant, the question most frequently used was whether or not it was located at the beginning of a word.

Table 1 MDL and ML performance.

	D	V	# of nodes	Recog. rate (%)
MDL	—	—	2,069	80.4
ML 1	60	0	3,739	75.4
ML 2	100	0	3,000	76.4
ML 3	200	0	2,001	76.7
ML 4	300	0	1,943	75.4
ML 5	400	0	1,200	73.4
ML 6	500	0	1,018	71.9
ML 7	1,000	0	591	66.6
ML 8	60	200	2,777	76.2
ML 9	60	400	2,034	77.0
ML 10	60	600	1,488	77.8
ML 11	60	800	1,248	77.9
ML 12	60	1,000	1,071	77.4
ML 13	200	200	1,782	77.3
ML 14	200	400	1,533	77.2
ML 15	200	600	1,326	77.0
ML 16	200	800	1,142	77.8
ML 17	200	1,000	1,049	76.5

Table 2 Distributions of questions asked.

Vowel		Consonant	
L-coronal	67	L-begin	130
L-dorsal	55	L-back	69
L-begin	40	R-a	63
R-coronal	39	R-high	62
L-h	35	L-high	60
L-back	34	L-a	53
L-sonorant	33	L-front	45
R-dorsal	31	R-e	34
L-unvoiced	27	R-back	32
L-n	27	L-e	30
L-fricative	27	L-consonant	28
Total	1,110	Total	822

Let us next consider how the optimal model size changes as the amount of data increases. Table 3 shows results for Data A and Data B. Data B, which is seven times larger than Data A, resulted in roughly a threefold increase in the number of nodes.

In order to evaluate the optimality of this size, we add a weight coefficient c to the second term on the

Table 3 Recognition rates (%) for Data A and Data B.

Training set	Data A	Data B
No. of nodes	2,069	6,223
Male 1	72.8	84.8
Male 2	76.8	84.4
Male 3	89.2	92.4
Male 4	81.6	83.6
Male 5	81.6	84.8
Average	80.4	86.0

Table 4 Recognition rates (%) as a function of coefficient c.

c	0.1	0.5	1.0	2.0	4.0	10.0
No. of nodes	13,927	9,798	6,223	3,949	2,418	1,341
Male 1	84.0	84.4	84.8	83.6	82.4	79.6
Male 2	81.6	83.6	84.4	84.4	84.8	80.8
Male 3	92.0	92.0	92.4	92.8	92.4	91.2
Male 4	84.8	85.2	83.6	85.2	84.8	82.0
Male 5	84.4	84.4	84.8	87.6	85.2	86.8
Average	85.4	85.9	86.0	86.7	85.9	84.1

Table 5 Recognition rates (%) with mixture-Gaussian output pdfs.

	1 Gauss	2 Gauss
Male 1	84.8	86.8
Male 2	84.4	87.6
Male 3	92.4	94.4
Male 4	83.6	88.8
Male 5	84.8	87.2
Average	86.0	89.0

right-hand side of (16), which results in:

$$l''(U)=\frac{1}{2}\sum_{m=1}^{M}\Gamma_m(K+K\log(2\pi)+\log|\Sigma_m|) + cKM\log W+C. \quad (20)$$

As c increases, so does the penalty for a large model size. Table 4 shows results for a range of c values of from 0.1 to 10.0. While the highest recognition accuracy was achieved for a c value of 2.0, it was only 0.7% higher than that for $c=1$ (*i.e.*, for the case in which the penalty for increased mode size is the same as that in (16), which expresses the description length in our approach.)

We also used Data B to evaluate the recognition performance when single-Gaussian output pdfs were replaced with mixture-Gaussian output pdfs. In this experiment, the number of Gaussian pdfs assigned to each state was increased to two, and the model was retrained using the same training data. The increase in recognition rates shown in Table 5, indicates that further splitting of some of the nodes in models constructed using the MDL criterion might result in improved recognition rate if a better set of questions were prepared beforehand. Such a set might include, for example, questions regarding second-to-left and/or second-to-right phones, characteristics of individual speakers, recording conditions, *etc.*

7. DISCUSSION

While we have proposed here a significantly useful method of optimizing the model size without any externally given parameters, there still remain a number of problems to be solved. First, we have yet to determine the degree to which the assumptions included in our method affect its performance in model size control. Second, the set of models provided beforehand may not include the most optimal model ("true model") for the given data. This is, of course, not only true for the proposed method, but also generally true for other model selection strategies using the MDL criterion, and further theoretical research is needed in order to address this problem. Third, minimization of the description length does not necessarily imply a minimization of recognition error. Conventional ML approaches encounter the same problem: maximization of likelihood does not necessarily mean the minimization of recognition error. The MDL criterion has an advantage over the ML criterion in that it has an effective penalty used for model size control, one that has good theoretical support.

8. CONCLUSION

We have proposed here a training method for acoustic modeling that generates HMMs with an appropriate model size. In an evaluation test, it achieved higher recognition accuracy than a conventional approach with much lower overall computational costs.

The MDL criterion can be applied not only to state splitting using phonetic decision trees but also to other clustering methods, such as agglomerative clustering. It can also be applied to discriminative training. Studies in such directions would seem to hold promise.

REFERENCES

1) K. -F. Lee, S. Hayamizu, H. -W. Hon, C. Huang, J. Swartz, and R. Weide, "Allophone clustering for continuous speech recognition," Proc. ICASSP90, Albuquerque, 749-753 (1990).
2) M. -Y. Hwang, X. Huang, and F. Alleva, "Predicting unseen triphones with senones," Proc. ICASSP93, Minneapolis, II-311-314 (1993).
3) V. Digalakis, P. Monaco, and H. Murveit, "Genons: Generalized mixture tying in continuous hidden Markov model-based speech recognizers," IEEE Trans. SAP **4**, 281-289 (1996).
4) S. J. Young, "The general use of tying in phoneme-based HMM speech recognizers," Proc. ICASSP92, San Francisco, 569-572 (1992).
5) J. Takami and S. Sagayama, "A successive state splitting algorithm for efficient allophone modeling," Proc. ICASSP92, San Francisco, I-573-576 (1992).
6) M. Ostendorf and H. Singer, "HMM topology design using maximum likelihood successive state splitting," Comput. Speech Lang. **11**, 17-41 (1997).
7) L. R. Bahl, P. V. de Souza, P. S. Gopalakrishnan, D. Nahamoo, and M. A. Picheny, "Decision trees for phonological rules in continuous speech," Proc. ICASSP91, Toronto, 185-188 (1991).
8) C. -H. Lee, E. Giachin, L. R. Rabiner, R. Pieraccini and A. E. Rosenberg, "Improved acoustic modeling for large vocabulary continuous speech recognition," Comput. Speech Lang. **6**, 103-207 (1992).
9) S. J. Young, J. J. Odell, and P. C. Woodland, "Tree-based state tying for high accuracy acoustic modeling," Proc. Hum. Lang. Technol., 307-312 (1994).
10) R. M. Schwartz, Y. Chow, S. Roucos, M. Kransner, and J. Makhoul, "Improved hidden Markov modeling of phonemes for continuous speech recognition," Proc. ICASSP 84, 35.6 (1984).
11) K. Shinoda and T. Watanabe, "Acoustic modeling based on the MDL criterion for speech recognition," EuroSpeech97 **1**, 99-102 (1997).
12) J. Rissanen, "Universal coding, information, prediction, and estimation," IEEE Trans. IT **30**, 629-636 (1984).
13) L. Rabiner and B. -H. Juang, *Fundamentals of Speech Recognition* (Prentice Hall, Englewood Cliffs, 1993).
14) A. Kannan, M. Ostendorf, and J. Rohlicek, "Maximum likelihood clustering of Gaussians for speech recognition," IEEE Trans. SAP **2**, 453-454 (1994).

Koichi Shinoda received his B. E. and M. E. degrees in physics from the University of Tokyo in 1987 and 1989, respectively. In 1989 he joined NEC Corp. Japan and was involved in research on automatic speech recognition. From 1997 to 1998 he was a visiting scholar in Bell Labs, Lucent Technologies. He received the Awaya Prize from Acoustic Society of Japan in 1997 and Excellent Paper Award from the Institute of Electronics, Information and Communication Engineers in 1998. His current research interests include speech recognition, statistical pattern recognition, and information theory. He is a member of IEICE and IEEE.

Takao Watanabe received his B. E. and M. E. degrees from the University of Tokyo in 1972 and 1974, respectively. He received his Dr. Eng. from Waseda University in 1994. Since 1974 he has been working on research and development on speech and language processing at NEC Corporation. He is Senior Manager of C&C Media Research Laboratories. He is a member of the Acoust. Soc. Japan, the Institute of Electronics, Information and Communication Engineers, and IEEE.