SPEAKER ADAPTATION OF THE SSS (SUCCESSIVE STATE SPLITTING)-BASED HIDDEN MARKOV NETWORK FOR CONTINUOUS SPEECH RECOGNITION

Jun-ichi Takami, Akito Nagai and Shigeki Sagayama

ATR Interpreting Telephony Research Laboratories

ABSTRACT – This paper describes a speaker adaptation method called Vector Field Smoothing (VFS) for a Hidden Markov Network (HMnet) generated by the Successive State Splitting (SSS) algorithm, and shows experimental results of speech recognition for multiple input speakers. The VFS method can accurately adapt a standard speaker's HMnet to the input speaker's HMnet with a limited amount of training samples because of its "smoothing" mechanism for transfer vectors. By using this method, remarkable improvements in the continuous speech recognition rates have been obtained.

INTRODUCTION

In our quest for accurate speech recognition, we are making a study of speech recognition methods using phoneme-context-dependent Hidden Markov Models (HMMs).

To model precise phoneme-context-dependent HMMs, we have developed a Successive State Splitting (SSS) algorithm (Takami et al., 1992). The SSS algorithm can simultaneously and automatically optimize all items of a model unit, the model architecture and the model parameters with a maximum likelihood criterion, and can generate an efficient network of phoneme-context-dependent HMMs called the Hidden Markov Network (HMnet). With this phoneme-context-dependent approach, the HMnet can achieve high speech recognition performance even when a single Gaussian density distribution is used for each output probability density distribution. Moreover, since each state of the HMnet is efficiently shared among several different allophone models, the total amount of free parameters of the HMnet is less than that of existing mixture Gaussian density phoneme-context-independent HMMs.

The above advantages of the HMnet are especially effective for model re-estimation such as speaker adaptation where the amount of training samples for adaptation is limited, making it difficult to adapt a lot of model parameters.

An effective speaker adaptation method called Vector Field Smoothing (VFS) has also been developed at ATR (Hattori et al., 1992, Ohkura et al., 1992). In this method, a speaker adaptation problem is formulated for elastic conversion of the model parameter field. Using this method, standard speaker's HMMs can be accurately adapted into the input speaker's HMMs with fewer training samples.

Recently, we tested the total performance of the HMnet for multiple speakers' utterances by applying the VFS method to the HMnet.

In this paper, we explain the principle of the HMnet and the SSS algorithm, the mechanism of the VFS method, and show the continuous speech recognition experimental results for multiple speakers obtained by combining the speaker-adapted HMnet and a phoneme-context-dependent LR parser.

PHONEME-CONTEXT-DEPENDENT PHONE MODELS

It is known that speech recognition using phoneme-context-dependent HMMs is an effective approach for achieving high speech recognition performance despite the variations in feature parameters due to differences in phoneme contexts (Sagayama, 1989, Lee et al., 1990). On the other hand, when the number of models becomes larger by classifying each phoneme context class into more precise divisions, the amount of free parameters increases. Therefore, it is difficult to stochastically estimate accurate phoneme-context-dependent HMMs with limited training samples. To overcome this problem, it is important to reduce as much as possible the number of useless free parameters from each model and to efficiently obtain the information on training samples with a smaller number of free parameters.

To generate accurate phoneme-context-dependent HMMs, we proposed a Successive State Splitting (SSS) algorithm (Takami et al., 1992). By using the maximum likelihood criterion, the SSS can simultaneously and automatically optimize the following three items which are important to construct phoneme-context-dependent HMMs:

- the model unit, i.e. the set of phoneme context classes,
- the model architecture, i.e. the number of states per model and the architecture of state sharing,
- the model parameters, i.e. output probability density distributions and state transition probabilities.

Using this algorithm, an efficient network of phoneme-context-dependent HMMs called the Hidden Markov Network (HMnet) is generated (Takami et al., 1992).

Below, the SSS algorithm and the HMnet are described briefly.

Successive State Splitting (SSS) Algorithm

The concept of SSS is to successively make each model more precise by iterating the split of a probabilistic statistical signal source, i.e. a hidden Markov state, into either a phoneme contextual domain or a temporal domain based on the maximum likelihood criterion.

However, to achieve this concept directly, it is necessary to evaluate all possible combinations. Specifically, this implies determining in which state and in which domain a split can achieve the maximum likelihood after actually generating all possible networks. Such a huge computation is not practical in terms of the present computer's capabilities.

Consequently, in the actual algorithm, the following two approximations have been introduced:

- at each iteration, the state having the largest output probability density distribution is determined as the state to split,
- the output probability density distribution of each state is formed as a two-mixture Gaussian density distribution, and when a state is split, each of the two Gaussian density distributions of the original state is distributed to one of two new states.

By these approximations, the state to split and the output probability density distributions of the new states can be determined without any training process. As a result, a great amount of computation has been reduced. Figure 1 shows the principle of SSS.

Hidden Markov Network (HMnet)

The HMnet is a network of multiple hidden Markov states, and each state has the following information:

- state index,
- acceptable contextual class,
- lists of preceding states and succeeding states,
- parameters of the output probability density distribution,
- state transition probabilities.

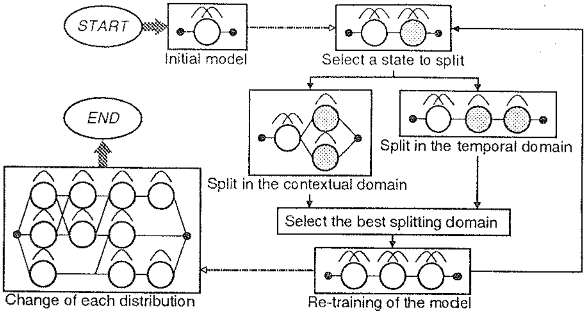
START
Initial model
Select a state to split
END
Split in the temporal domain
Split in the contextual domain
Select the best splitting domain
Change of each distribution
Re-training of the model

Figure 1: The Successive State Splitting algorithm

In the HMnet, if a phoneme context of a sample is given, the model corresponding to the context can be determined by concatenating several states, each of which can accept the context, by applying the restrictions of the preceding state list and the succeeding state list. Since this model is equivalent to a common HMM, we can use the forward-pass algorithm to calculate the likelihoods for input samples as well as for common HMMs.

SPEAKER ADAPTATION BY VECTOR FIELD SMOOTHING METHOD

Since each output probability density distribution of the HMnet is a single Gaussian density distribution, it can be expected to achieve effective speaker adaptation even using a very simple method, e.g. the re-estimation of only the mean vector of each Gaussian density distribution.

In actuality, however, the mean vector of a state corresponding to each phoneme-context not appearing in the training samples can not be adapted. Moreover, adapted mean vectors include some estimation errors resulting from a small amount of training samples.

To overcome this problem, a speaker adaptation method called Vector Field Smoothing (VFS) was developed at ATR (Hattori et al., 1992) (Ohkura et al., 1992). This method can simultaneously perform both estimation of the mean vectors not adapted due to the lack of corresponding training samples and correction of the estimation errors of adapted mean vectors by transfer vector smoothing with a spatial filter.

The VFS algorithm consists of the following two steps:

1. calculation of transfer vectors,
2. smoothing of transfer vectors.

Below, the details of each step are shown.

Calculation of Transfer Vectors

In the first step, to obtain the adapted mean vectors, the mean vectors of each output probability density distribution are re-estimated by embedded training with the input speaker's training samples and the information from their phonetic transcription.

The embedded training is performed using the Baum-Welch algorithm. In this training, only mean vectors are re-estimated by fixing both the variance vectors and the state transition probabilities.

Here, the transfer vector for the output probability density distributions of state i is calculated by Eq.1.

$$\Delta\vec{\mu}_i = \vec{\mu}_i{}' - \vec{\mu}_i \quad , \tag{1}$$

where,

i : the state index ,
$\Delta\vec{\mu}_i$: the transfer vector of the distribution of state i
(if the distribution of state i has not been adapted, $\Delta\vec{\mu}_i$ does not exist) ,
$\vec{\mu}_i$: the mean vector of the original distribution of state i ,
$\vec{\mu}_i{}'$: the mean vector of the adapted distribution of state i .

Smoothing of Transfer Vectors

In the next step, the transfer vectors for the adapted mean vectors are smoothed using a spatial filtering technique. This is to estimate mean vectors that have not been adapted since their corresponding phoneme context had not appeared in the training samples, and to correct the estimation errors of mean vectors resulting from a small amount of training samples.

The smoothing is performed using Eq.2, and the final mean vector $\hat{\vec{\mu}}_i$ for every state i can be calculated.

$$\begin{aligned} \hat{\vec{\mu}}_i &= \vec{\mu}_i + \sum_{k=1}^{K} \Delta\vec{\mu}_{c(k)} w_{ic(k)} \quad , \\ w_{ij} &= e^{-d_{ij}/\lambda} \Big/ \sum_{k=1}^{K} e^{-d_{ic(k)}/\lambda} \quad , \\ d_{ij} &= \sum_{l=1}^{L} (\mu_{il} - \mu_{jl})^2 / \sigma_{il}^2 \quad , \end{aligned} \tag{2}$$

where,

$\hat{\vec{\mu}}_i$: the final mean vector of the distribution of state i ,
K : the number of neighborhoods ,
$c(k)$: the $k-th$ neighbor state index of state i
(if $\Delta\vec{\mu}_j$ does not exist, j is not included in the candidates of $c(k)$) ,
λ : the smoothing rate ,
L : the parameter dimension ,
μ_{il} : the $l-th$ mean value of the original distribution of state i ,
σ_{il}^2 : the $l-th$ variance value of the original distribution of state i .

The smoothing rate λ is a variable to control the strength of the smoothing, and corresponds to the window width of the spatial filter. Stronger smoothing is possible by increasing λ.

CONTINUOUS SPEECH RECOGNITION EXPERIMENTS

ATREUS/SSS-LR Continuous Speech Recognition System

To test the performance of our methods, continuous speech recognition experiments were carried out using our continuous speech recognition system called "ATREUS/SSS-LR" (Nagai et al., 1992). This system is constructed by combining an HMnet-based phoneme verifier and a phoneme-context-dependent LR parser.

The phoneme-context-dependent LR parser is an effective parsing algorithm also used at ATR for driving phoneme-context-dependent phone models (Nagai et al., 1991). This parser can efficiently grow the parsing tree by predicting phonetic triplet hypotheses one after another according to the grammar constraints.

This system has been found to achieve the highest performance for speaker-dependent phrase recognition among many systems developed at ATR.

Another advantage of this system is that the recognition processing can be performed with a small amount of computation. Since the HMnet represents accurately various acoustic fluctuations caused due to the differences in the preceding or succeeding phonemes, i.e. phoneme contexts, the correct candidates tend constantly to be higher in rank in the parsing tree. Therefore, high recognition rates can be obtained with a smaller beam width in the beam search process than that used in the phoneme-context-independent approaches. Moreover, the output probability for each distribution itself can be calculated with a small amount of computation because of both the simplicity of each output probability density distribution and the small amount of total free parameters.

The continuous speech recognition experiments for multiple input speakers noted in this section were performed applying the aforementioned speaker adaptation method to the HMnet used in the existing "ATREUS/SSS-LR" speaker-dependent continuous speech recognition system.

Experimental Conditions

Table 1 shows the experimental conditions.

Table 1: Experimental conditions

recognition task	conversational sentences uttered phrase by phrase
input speakers	2 males (MHT, MXM) and 1 female (FSU)
standard speaker	1 male (MAU)
acoustic analysis	12kHz–16bit sampling, 20ms Hamming window, 5ms frame period
feature parameters	34 dimensional vector (*logpow*+*cep*(16)+Δ*logpow*+Δ*cep*(16))
samples for adaptation	top N words of phonetically balanced 216 words (N : 10, 25, 50, 100)
state number of HMnet	600 (1688 different allophonic HMMs are represented)
neighborhood count	6
smoothing rate λ	10.0, 20.0, 30.0
grammar	1035 words, 1407 rules, phoneme perplexity 5.9
beam width	256

Recognition Experimental Results

Table 2 shows the recognition experimental results. In this table, the phrase recognition rates are obtained using the smoothing rate λ capable of achieving the best recognition rates for all speakers in average.

By using the VFS speaker adaptation method, remarkable improvements in recognition performance were achieved even when the differences in the feature parameters between speakers seemed to be large, such as between male speaker MAU and female speaker FSU.

It was also found that the smoothing rate λ capable of achieving the highest recognition rates is large when there is a relatively small amount of training samples, and is small when there is a relatively large amount of training samples. This tendency is reasonable from the viewpoint of estimation error correction by transfer vector smoothing.

Table 2: Japanese phrase recognition results for multiple speakers.

#training samples N	recognition rates for top 1 choice (%)							
	input speakers							standard speaker
	with adaptation				without adaptation			
	λ	MHT	MXM	FSU	MHT	MXM	FSU	MAU
10 words	30.0	84.06	87.36	57.19	47.10	73.29	3.96	93.19
25 words	20.0	88.41	87.36	65.11				
50 words	10.0	90.94	88.09	84.53				
100 words	10.0	92.39	87.36	89.21				

These results confirm that the VFS principle is an effective method for speaker adaptation with a small amount of training samples.

CONCLUSIONS

In this paper, we described a speaker adaptation method called Vector Field Smoothing (VFS) for a Hidden Markov Network (HMnet) generated by the Successive State Splitting (SSS) algorithm, and showed experimental results of speech recognition for multiple input speakers.

Through continuous speech recognition experiments using the "ATREUS/SSS-LR" continuous speech recognition system, we showed the high performance of our speaker adaptation method.

ACKNOWLEDGMENTS

We would like to thank Dr. A. Kurematsu, President, ATR Interpreting Telephony Research Laboratories, for his continuous support of this work. We also acknowledge all the members of the Speech Processing Department for their discussions and encouragements.

REFERENCES

Hattori, H. and Sagayama, S. (1992) *Vector Field Smoothing Principle for Speaker Adaptation*, Proc. ICSLP'92, We.fPM.1.4, to appear.

Lee, K.F. , Hayamizu, S. , Hon, H.W. , Huang, C. , Swartz, J. and Weide, R. (1990) *Allophone Clustering for Continuous Speech Recognition*, Proc. ICASSP'90, 749-752.

Nagai, A. , Sagayama, S. and Kita, K. (1991) *Phoneme-context-dependent LR Parsing Algorithms for HMM-based Continuous Speech Recognition*, Proc. Eurospeech'91, 1397-1400.

Nagai, A. , Takami, J. and Sagayama, S. (1992) *The SSS-LR Continuous Speech Recognition System: Integrating SSS-Derived Allophone Models and a Phoneme-Context-Dependent LR Parser*, Proc. ICSLP'92, Fr.AM.P, to appear.

Ohkura, K. , Sugiyama, M. and Sagayama, S. (1992) *Speaker Adaptation based on Transfer Vector Field Smoothing with Continuous Mixture Density HMMs*, Proc. ICSLP'92, We.fPM.1.1, to appear.

Sagayama, S. (1989) *Phoneme Environment Clustering for Speech Recognition*, Proc. ICASSP'89, 397-400.

Takami, J. and Sagayama, S. (1992) *Successive State Splitting Algorithm for Efficient Allophone Modeling*, Proc. ICCASP'92, 573-576.