FEATURE-DEPENDENT ALLOPHONE CLUSTERING

Shigeki Matsuda, Mitsuru Nakai, Hiroshi Shimodaira, and Shigeki Sagayama

Japan Advanced Institute of Science and Technology
Tatsu-no-Kuchi, Ishikawa, 923-1292 Japan
E-mail: {matsuda,mit,sim,sagayama}@jaist.ac.jp

Abstract

We propose a novel method for clustering allophones called Feature-Dependent Allophone Clustering (FD-AC) that determines feature-dependent HMM topology automatically. Existing methods for allophone clustering are based on parameter sharing between the allophone models that resemble each other in behaviors of feature vector sequences. However, all the features of the vector sequences may not necessarily have a common allophone clustering structures. It is considered that the vector sequences can be better modeled by allocating the optimal allophone clustering structure to each feature. In this paper, we propose Feature-Dependent Successive State Splitting (FD-SSS) as an implementation of FD-AC. In speaker-dependent continuous phoneme recognition experiments, HMMs created by FD-SSS reduced the error rates by about 10% compared with the conventional HMMs that have a common allophone clustering structure for all the features.

1. INTRODUCTION

In recent speech recognition techniques, hidden Markov models (HMMs) are one of the most powerful techniques for modeling the time sequential data. HMM models non-stationary feature vector sequences by switching the finite number of stationary vector signal sources. However, being limited amount of training data available, more than enough number of free parameters leads to over-fitting or over-learning and resulting in poor generalization. Hence it is essential to devise an efficient representation scheme of acoustic features with less number of free parameters. Parameter tying is a typical manner that reduces the excess free parameters. The parameter tying techniques can be classified generally into the following four levels.

Level 1: allophone clustering [1, 2]
Level 2: state tying [1, 2, 3]
Level 3: tied mixtures [4]
Level 4: distribution parameter tying [5]

The allophone clustering shares the parameters of the allophone models that are similar each other in the behaviors of feature vector sequences. The total number of parameters can be reduced efficiently with minimum loss of information by the tying technique. A number of clustering algorithms have been proposed in this framework. But none has tried to relax the tacit constraint on acoustic features, i.e., treating the features as a vector. In speech recognition, feature vector sequences often consist of MFCCs, a power and its time derivatives. Since these individual features have different behaviors and different features may have different context dependencies, all the features may not necessarily have a common allophone clustering structure. It is considered that the feature vector sequences can be modeled more efficiently by clustering allophones for each feature separately.

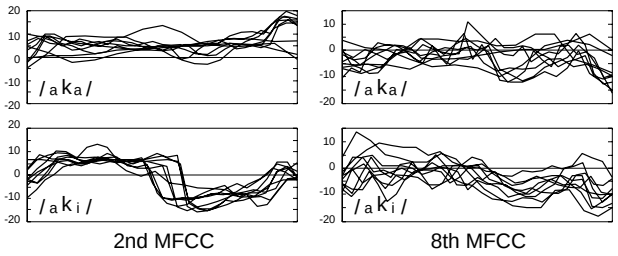

Figure. 1: Trajectories of 2nd and 8th MFCC of allophones /aka/ and /aki/

In this paper, we propose Feature-Dependent Allophone Clustering (FD-AC) that clusters allophones for each feature separately. In other word, FD-AC determines the tying structure of distribution parameters for each feature separately. In section 2, we describe a principle of FD-AC and Feature-Dependent Successive State Splitting as an implementation of FD-AC. In section 3, HMMs with different allophone clustering structures created by FD-SSS are evaluated in the continuous phoneme recognition experiment. The last section is a conclusion.

2. FEATURE-DEPENDENT ALLOPHONE CLUSTERING

In speech recognition, MFCCs and their time-derivatives (delta MFCCs) are commonly used as acoustic feature vectors. It is generally considered that the lower-order MFCCs have more significant information than the higher-order MFCCs. Therefore, assigning more number of HMM states to the lower-order MFCCs than to the higher-order MFCCs may result in better recognition performance. Fig. 1 shows an example of feature vector sequences of a phoneme /aka/ and a phoneme /aki/. Here, /aki/ denotes the phoneme /k/ with the preceding phoneme /a/ and the following phoneme /i/. We can see that the both phonemes have similar trajectories for the 8th MFCC, but different for the 2nd MFCC. In this case, it is better to assign a common HMM state to the 8th MFCC of those two allophones so that the number of free HMM parameters is reduced, while, for the 2nd MFCC, they

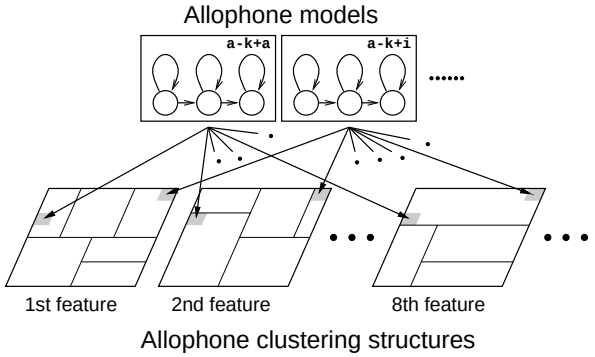

Figure. 2: Feature-Dependent Allophone Clustering (FD-AC) that creates different allophone clustering structures for individual features

need different HMM states each other. Therefore, allocating the optimal number of HMM free parameters and determining the tying structure for each individual feature by means of clustering will lead to a better modeling that shows higher precisions and robustness than the existing methods.

Fig. 2 illustrates the idea of Feature-Dependent Allophone Clustering. In this figure, each feature space consists of different number of clusters: there are six clusters in the 1st feature, five clusters in the 2nd feature and four clusters in the 8th feature. Allophones /aka/ and /aki/ share a common state only for the 8th feature.

2.1. Feature-Dependent Successive State Splitting

In order to obtain the feature-dependent allophone clusters and to decide the optimal numbers of free HMM parameters, we extend Successive State Splitting (SSS)[2] that is one of the most popular algorithms for creating the hidden Markov networks (HMnet). The SSS algorithm increases the model likelihood successfully by splitting a state in HMnet that represents a state tying structure and an allophone clustering structure. We utilize the SSS algorithm for the proposed feature-dependent allophone clustering and call the new algorithm as FD-SSS. There are two possible implementations of FD-SSS as follows.

Synchronous type: Create a HMnet for each feature with the constraint that the state transition of each feature's HMnet takes place synchronously with those of other feature's HMnets (Fig. 3).

Asynchronous type: Create a HMnet for each individual feature separately, and combine the HMnets into the Asynchronous-Transition HMMs [6, 7], in which state transition occurs asynchronously for all the features but the same efficient decoding algorithm such as one-pass Viterbi search can be still applied with the conventional HMMs.

In this paper, we discuss only the asynchronous type. The outline of the FD-SSS algorithm is shown in Fig. 4.

Step 1: Train a single state HMM (HMnet) for each feature with all the training phone samples, where the output probability for each feature is modeled by a single Gaussian distribution with two scalar parameters, i.e., a mean and a variance.

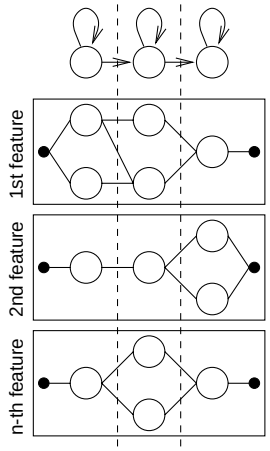

Figure. 3: FD-SSS (synchronous type) for generating a FD-HMnet. Each vertical dotted-line indicates the synchronized state-transitions for all the features.

Step 2: Among the states of all the HMnets (i.e., HMnet for the 1st feature through the HMnet for the n-th feature), find the one that will earn the largest likelihood gain when being split into two states. The state splitting gains are examined both in contextual and temporal domains.

Step 3: Split the state and re-train all the states that are affected by the split using the corresponding data subsets.

Step 4: Repeat steps 2 and 3 until the number of all states reaches a preset number.

Step 5: Create AT-HMMs [6, 7] with the allophone clustering structures of the obtained HMnets by clustering the state transition timings.

The state splitting gain G in **Step 2** can be calculated by the same equation used in ML-SSS. In case of splitting state s into two states, q_1 and q_2, the gain is defined by

$$\begin{aligned} G(s,q_1,q_2) &= -N_2(s,s)\log a_{ss} + 0.5N_1(s)\log\sigma(s) \\ &\quad + \sum_{s'=q_1q_2}\sum_{s''=q_1q_2} N_2(s',s'')\log a_{s's''} \\ &\quad - \sum_{s'=q_1q_2} 0.5N_1(s')\log\sigma(s'), \qquad (1) \\ N_1(s') &= \sum_t \gamma_t(s'), \qquad (2) \\ N_2(s',s'') &= \sum_t \xi_t(s',s''), \qquad (3) \end{aligned}$$

where $\sigma(s)$ is the variance of state s, $\gamma_t(s')$ represents the probability of being in state s' at time t, and $\xi_t(s',s'')$ denotes the probability of being in state s' at time t and state s'' at time $t+1$.

3. EXPERIMENTS

To evaluate the proposed method, the ATR word speech database of Japanese important 5240 words uttered by 2 male speakers (MHT, MAU) and 2 female speakers (FMS, FFS) was used. Out of them, the odd-numbered 2620 words and the phoneme bal-

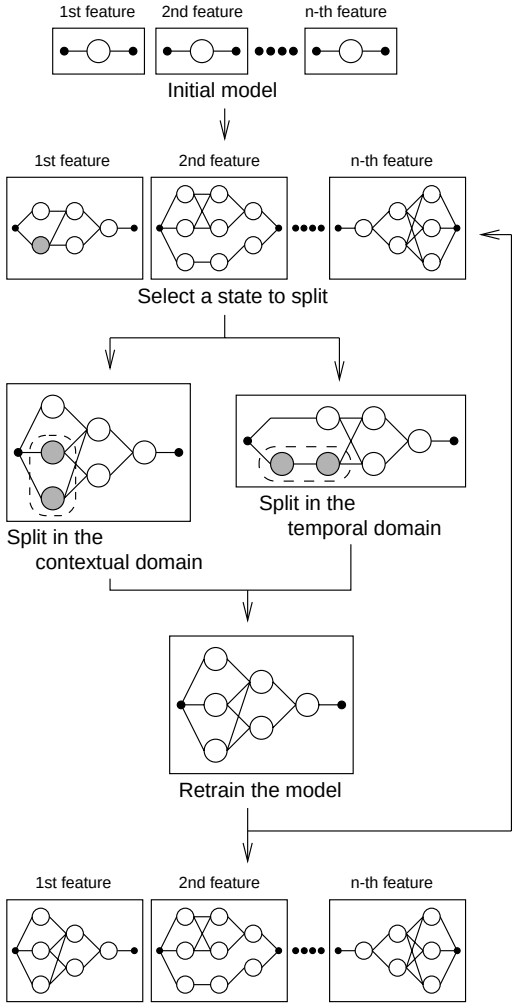

Figure. 4: FD-SSS (asynchronous type) for generating a FD-HMnet with the optimal state tying structure

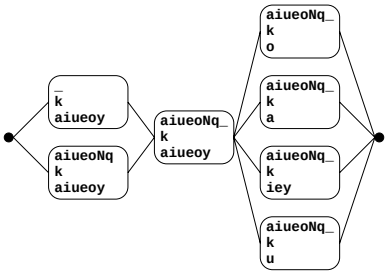

Figure. 5: Example of the HMnet topologies by ML-SSS for /k/

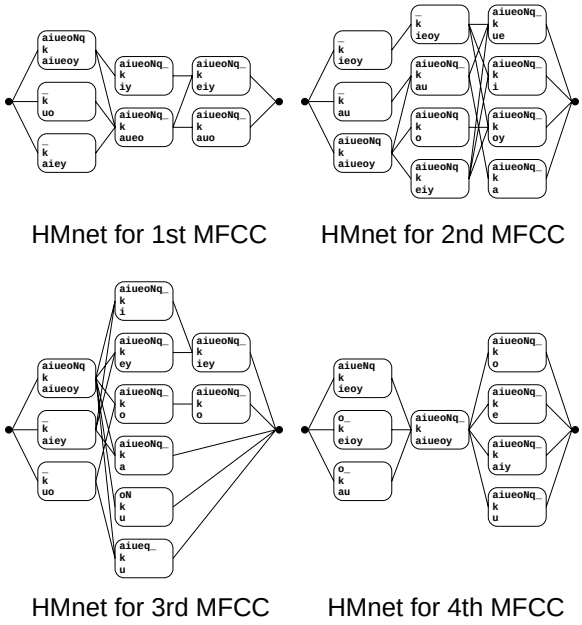

Figure. 6: Examples of the HMnet topologies by FD-SSS for /k/

anced 216 words were used for training the speaker-dependent HMnet, and half of the even-numbered 1310 words were used for testing. 12 MFCCs, 12 ΔMFCCs, log-power and Δlog-power extracted with 5ms frame period and 25ms frame length were used as an acoustic feature vector. The phoneme categories for recognition were / N, a, b, ʧ, d, e, f, g, h, i, ʒ, k, m, n, o, p, q, r, s, ʃ, t ʦ, u, w, j, z /.

3.1. Model Generation using FD-SSS

Fig. 5 and Fig. 6 show the examples of the HMnet topologies for phoneme /k/ of a male speaker (MHT), where the former (Fig. 5) was created by ML-SSS, while the latter (Fig. 6) was built by the proposed FD-SSS algorithm. The upper row in each round box denotes the left context of the phoneme shown in the middle row, and the lower row represents the right context. As is shown in Fig. 6, the HMnets by FD-SSS (FD-HMnet) have different state-tying structures for different features, which supports our assumption that different features may have different allophone clustering structures.

Fig. 7 shows the number of states that were assigned to each individual feature in the FD-HMnet under the condition that total number of distribution parameters is fixed to 20000. It can be seen from the figure that the numbers of assigned states are different among the features, implying that the proposed FD-SSS successfully allocated the optimal number of free-parameters to the allophone structure for each individual feature in the sense of maximum-likelihood estimation.

Fig. 8 illustrates the average variances of the output distributions of the FD-HMnet in comparison with those of ML-SSS based HMnet (ML-HMnet). We can see that the FD-HMnet has smaller variances than the ML-HMnet. It is considered that the FD-HMnet represents the information of the training data more efficiently than ML-HMnet.

3.2. Continuous Phoneme Recognition

For the evaluation of FD-HMnet generated by FD-SSS, continuous phoneme recognition experiments were performed in comparison with the conventional ML-HMnet created by ML-SSS. For decoding, the one-pass Viterbi algorithm was used with the phonotactic constraints in Japanese. The number of states for phoneme HMM was fixed to 3 or 5, and the SSS algorithm was

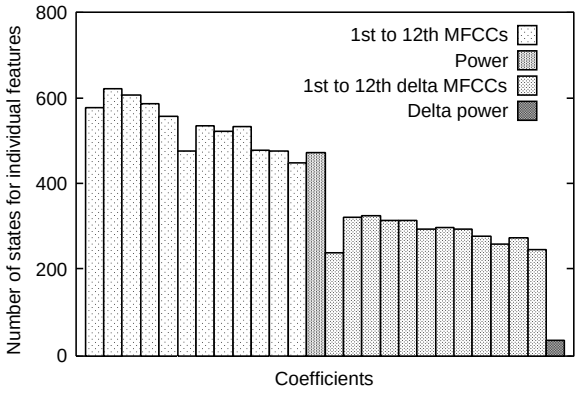

Figure. 7: Numbers of states of FD-HMnet created by FD-SSS

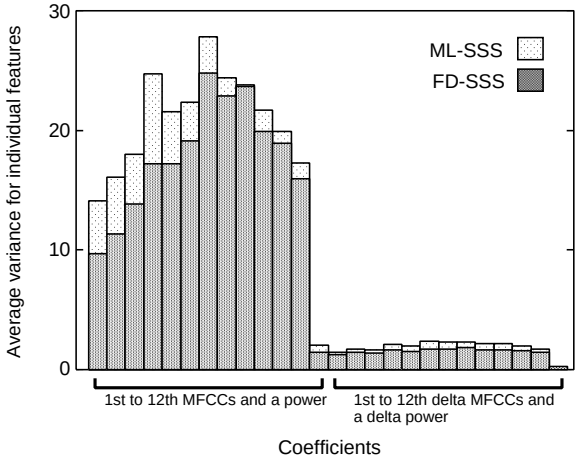

Figure. 8: Average variances of models created by ML-SSS and FD-SSS

Table. 1: Phoneme error rates of models generated by FD-SSS and ML-SSS

#parameters	#states	method	%error	%reduction
10400	3	ML-SSS	7.8	–
		FD-SSS	6.6	15.4
	5	ML-SSS	6.4	–
		FD-SSS	5.4	15.6
20800	3	ML-SSS	6.3	–
		FD-SSS	5.4	14.3
	5	ML-SSS	5.3	–
		FD-SSS	4.8	9.4

stopped when a number of free parameters reached 10400 or 20800.

Table 1 shows the experimental results where %reduction denotes the reduction of phoneme error rates defined by

$$\frac{\text{\%error of ML-SSS} - \text{\%error of FD-SSS}}{\text{\%error of ML-SSS}}.$$

We can see that FD-HMnet created by FD-SSS achieved a phoneme error reduction rate of about 10%. It is considered that acoustic feature vector sequences are modeled efficiently by FD-SSS based on FD-AC.

4. CONCLUSION

In this paper, we have proposed a new notion of feature-dependent allophone clustering (FD-AC) that clusters the allophones for each individual feature separately. As an implementation of FD-AC, we have developed Feature-Dependent Successive State Splitting (FD-SSS), in which SSS-like state splitting occurs separately for each individual feature to create a feature-dependent hidden Markov network (FD-HMnet).

In the continuous phoneme recognition experiments, the proposed FD-HMnet successfully reduced the error rates by about 10% compared with the conventional HMnet created by ML-SSS. It is considered that FD-AC is an effective technique for improving the speech recognition performance of the acoustic models.

Future works will include the evaluation of a speaker-independent model and the application to continuous speech recognition system.

REFERENCES

1. J. Takami and S. Sagayama, "A Successive State Splitting Algorithm for Efficient Allophone Modeling," in *Proc. ICASSP*, vol. I, pp. 573–576, 1992.
2. M. Ostendorf and H. Singer, "HMM Topology Design Using Maximum Likelihood Successive State Splitting," *Computer Speech and Language*, vol. 11, no. 1, pp. 17–41, 1997.
3. X. D. Huang, K. F. Lee, H. W. Hon, and M. Y. Hwang, "Improved Acoustic Modeling with the SPHINX Speech Recognition System," in *Proc. ICASSP*, vol. 1, pp. 345–248, 1991.
4. J. Bellegarda and D. Nahamoo, "Tied mixture continuous parameter models for large vocabulary isolated speech recognition," in *Proc. ICASSP*, vol. 1, pp. 13–16, 1989.
5. S. Takahashi and S. Sagayama, "Four-Level Tied-Structure for Efficient Representation of Acoustic Modeling," in *Proc. ICASSP*, vol. 1, pp. 520–523, 1995.
6. S. Matsuda, S. Sagayama, M. Nakai, and H. Shimodaira, "Asynchronous Transition HMM," in *Proc. ICASSP*, vol. 2, pp. 1005–1008, 2000.
7. S. Sagayama, S. Matsuda, M. Nakai, and H. Shimodaira, "Asynchronous Transition HMM for Acoustic Modeling," in *Proc. IEEE Workshop on Automatic Speech Recognition and Understanding*, 1999.