PAPER

A Fully Consistent Hidden Semi-Markov Model-Based Speech Recognition System

Keiichiro OURA[†a)], **Heiga ZEN**[†], ***Nonmembers***, **Yoshihiko NANKAKU**[†], **Akinobu LEE**[†], ***and*** **Keiichi TOKUDA**[†], ***Members***

SUMMARY In a hidden Markov model (HMM), state duration probabilities decrease exponentially with time, which fails to adequately represent the temporal structure of speech. One of the solutions to this problem is integrating state duration probability distributions explicitly into the HMM. This form is known as a hidden semi-Markov model (HSMM). However, though a number of attempts to use HSMMs in speech recognition systems have been proposed, they are not consistent because various approximations were used in both training and decoding. By avoiding these approximations using a generalized forward-backward algorithm, a context-dependent duration modeling technique and weighted finite-state transducers (WFSTs), we construct a fully consistent HSMM-based speech recognition system. In a speaker-dependent continuous speech recognition experiment, our system achieved about 9.1% relative error reduction over the corresponding HMM-based system.

key words: *speech recognition, hidden Markov model, hidden semi-Markov model, weighted finite-state transducer*

1. Introduction

Hidden Markov models (HMMs) (Fig. 1 (a)) have formed the basis of many speech recognition systems since the 1970s. The advantages of using HMMs are: i) They can represent speech as probability distributions. ii) They are robust to temporal structure variations. iii) They provide efficient algorithms for estimating their model parameters. However, a number of limitations of HMMs for modeling speech have been reported [1]. One of their major limitations is in duration modeling. In HMMs, state duration probabilities are implicitly modeled by state transition probabilities; state duration probabilities decrease exponentially with time. Geometric distribution calculated by the state transition probabilities of HMMs would be inappropriate state duration probability distribution representation of the temporal structure of speech.

One of the solutions to this problem is to integrate state duration probability distributions explicitly into the HMM. This model is known as a hidden semi-Markov model (HSMM) [2]–[4], and is illustrated in Fig. 1 (b). Unlike HMMs, HSMMs have state duration probability distributions. Figure 2 shows the state duration probability distributions of an HMM and an HSMM. Geometric distribution in an HMM would be inappropriate representation of the temporal structure of speech. Although the gamma, Poisson and log Gaussian distributions have been applied to state duration modeling in HMM-based speech recognition, in this paper, we assume that each state duration probability distribution is represented by a Gaussian distribution because

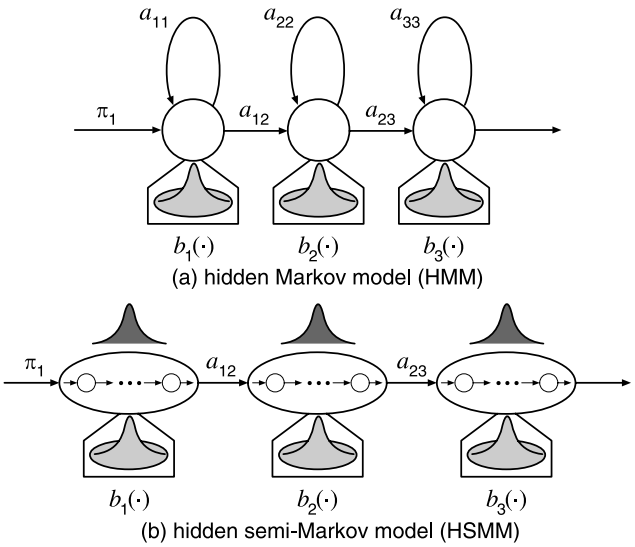


Fig. 1 Examples of a 3-state left-to-right HMM and an HSMM with no skip.

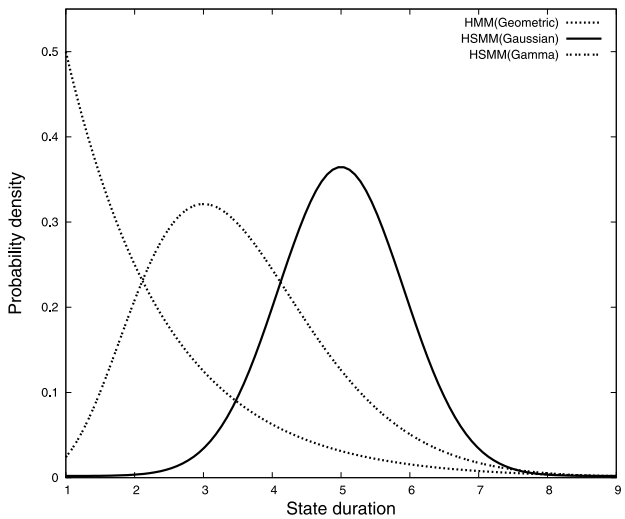


Fig. 2 State duration probability distributions.

Manuscript received March 14, 2008.
Manuscript revised July 4, 2008.

[†]The authors are with the Department of Computer Science and Engineering, Nagoya Institute of Technology, Nagoya-shi, 466–8555 Japan.
a) E-mail: uratec@sp.nitech.ac.jp

DOI: 10.1093/ietisy/e91–d.11.2693

there exists a simple clustering algorithm for Gaussian distributions. Although the clustering algorithm of the gamma distributions was reported in [5], we choose Gaussian distributions for simplicity in this paper.

Although discrete probability distributions can represent any distributions, the lack of training data could be an issue. In [4], a smoothing technique is used to prevent over training caused by the lack of training data. On the other hand, the use of continuous probability distributions may remedy such a over training problem. However, it is nessessary to choose an appropriate continuous probability distribution type previously. In this paper, we use continuous probability distributions because it can avoid additional processing such as smoothing.

A variety of attempts to include explicit duration models in speech recognition systems have been reported [6]–[8]. However, they are not fully consistent because various approximations are used in both training and decoding:

1) State duration probability distributions were estimated from statistical variables calculated by the forward-backward algorithm of the HMM, not of the HSMM [9].
2) State duration probability estimation utilizes a context-independent model or context-dependent state tying structure of state output probability distributions [7].
3) State duration models were not applied directly in the decoding process. Instead, the N-best hypotheses generated by the HMMs were rescored [8].

We propose a fully consistent HSMM-based speech recognition system to overcome the above approximations. For approximation 1), we simultaneously estimate both state output and duration probability distributions based on the HSMM statistics calculated by the generalized forward-backward algorithm [2]. For approximation 2), state output and duration probability distributions are independently clustered using different phonetic decision trees [9] by a decision-tree-based state clustering technique [10]. For approximation 3), we design an HSMM-native speech decoder based on weighted finite-state transducers (WFSTs) to apply HSMM directly to the input speech.

The rest of this paper is organized as follows. Section 2 describes training algorithms for the HSMM, context-clustering for state duration probability distributions, and a speech decoder for the HSMM using WFSTs. Results of our speech recognition experiment are presented in Sect. 3. Finally, concluding remarks and future plans are presented in Sect. 4.

2. A Fully Consistent HSMM-Based Speech Recognition System

2.1 Training Algorithms for HSMMs

We derived training algorithms for HSMMs based on the maximum likelihood (ML) criterion [9]. However, there is an inconsistency: state duration probability distributions have not been incorporated into the expectation step of the EM algorithm. In this section, the generalized forward-backward algorithm (expectation step) and parameter re-estimation formulas (maximization step) that are required to avoid the approximation in training [2], [4], [6], are described.

2.1.1 Generalized Forward-Backward Algorithm

The output probability of an observation vector sequence $\boldsymbol{o}$ from an HSMM Λ can be computed efficiently using the generalized forward-backward algorithm. The partial forward probabilities $\alpha_t(\cdot)$ and partial backward probabilities $\beta_t(\cdot)$ are defined as follows:

$$\alpha_0(j) = \begin{cases} 1 & j = N \\ 0 & \text{otherwise} \end{cases}, \tag{1}$$

$$\begin{aligned} \alpha_t(j) &= P(\boldsymbol{o}_1, \ldots, \boldsymbol{o}_t, q_t = j \mid q_{t+1} \neq j, \Lambda) \\ &= \sum_{d=1}^{t} \sum_{\substack{i=1, \\ i \neq j}}^{N} \alpha_{t-d}(i)\, a_{ij} p_j(d) \prod_{s=t-d+1}^{t} b_j(\boldsymbol{o}_s) \\ &\quad \left(\begin{matrix} t = 1, 2, \ldots, T \\ 1 \leq j \leq N \end{matrix} \right), \end{aligned} \tag{2}$$

$$\beta_{T+1}(i) = \begin{cases} 1 & i = N \\ 0 & \text{otherwise} \end{cases}, \tag{3}$$

$$\beta_T(i) = a_{iN} \beta_{T+1}(N) \quad (1 \leq i \leq N), \tag{4}$$

$$\begin{aligned} \beta_t(i) &= P(\boldsymbol{o}_{t+1}, \ldots, \boldsymbol{o}_T, q_t = i \mid q_{t+1} \neq i, \Lambda) \\ &= \sum_{d=1}^{T-t} \sum_{\substack{j=1, \\ j \neq i}}^{N} a_{ij} p_j(d) \prod_{s=t+1}^{t+d} b_j(\boldsymbol{o}_s) \beta_{t+d}(j) \\ &\quad \left(\begin{matrix} t = T-1, \ldots, 1 \\ 1 \leq i \leq N \end{matrix} \right), \end{aligned} \tag{5}$$

where a_{ij}, $b_j(\boldsymbol{o}_t)$, N, and $p_j(d)$, are a state transition probability from the i-th state to the j-th state, a state output probability of an observation vector $\boldsymbol{o}_t$ from the j-th state, the total number of states, and a state duration probability of the j-th state, respectively. From the above equations, the output probability of the observation vector sequence $\boldsymbol{o} = \{\boldsymbol{o}_1, \ldots, \boldsymbol{o}_T\}$ from the HSMM Λ is given by

$$\begin{aligned} P(\boldsymbol{o} \mid \Lambda) &= \sum_{i=1}^{N} \sum_{\substack{j=1, \\ j \neq i}}^{N} \sum_{d=1}^{t} \alpha_{t-d}(i)\, a_{ij} p_{j(d)} \prod_{s=t-d+1}^{t} b_{j(\boldsymbol{o}_s)} \beta_{t(j)} \\ &\quad (t = 1, \ldots, T). \end{aligned} \tag{6}$$

2.1.2 Parameter Re-Estimation Formulas

In this paper, we assume that each state output probability $b(\cdot)$ is represented by a mixture of Gaussian distributions. Parameter re-estimation formulas of the mixture weight w_{jg}, mean vector $\boldsymbol{\mu}_{jg}$ and covariance matrix $\boldsymbol{\Sigma}_{jg}$ of the g-th mixture of the j-th state are given by

$$\overline{w}_{jg} = \frac{\sum_{t=1}^{T}\sum_{d=1}^{t}\gamma_t^d(j,g)}{\sum_{h=1}^{G}\sum_{t=1}^{T}\sum_{d=1}^{t}\gamma_t^d(j,h)}, \tag{7}$$

$$\overline{\boldsymbol{\mu}}_{jg} = \frac{\sum_{t=1}^{T}\sum_{d=1}^{t}\zeta_t^d(j,g)}{\sum_{t=1}^{T}\sum_{d=1}^{t}\gamma_t^d(j,g)}, \tag{8}$$

$$\overline{\boldsymbol{\Sigma}}_{jg} = \frac{\sum_{t=1}^{T}\sum_{d=1}^{t}\eta_t^d(j,g)}{\sum_{t=1}^{T}\sum_{d=1}^{t}\gamma_t^d(j,g)}, \tag{9}$$

respectively, where G is the number of Gaussian distributions, $\gamma_t^d(j,g)$, $\zeta_t^d(j,g)$ and $\eta_t^d(j,g)$ are occupancy probabilities, first, and second order statistics, respectively, given by

$$\gamma_t^d(j,g) = \frac{1}{P(\boldsymbol{o}\mid\Lambda)}\sum_{\substack{i=1,\\ i\neq j}}^{N}\alpha_{t-d}(i)\,a_{ij}p_j(d)\beta_t(j) \cdot \sum_{s=t-d+1}^{t} w_{jg}\mathcal{N}\left(\boldsymbol{o}_s\mid\boldsymbol{\mu}_{jg},\boldsymbol{\Sigma}_{jg}\right) \cdot \prod_{\substack{k=t-d+1,\\ k\neq s}}^{t} b_j(\boldsymbol{o}_k), \tag{10}$$

$$\zeta_t^d(j,g) = \frac{1}{P(\boldsymbol{o}\mid\Lambda)}\sum_{\substack{i=1\\ i\neq j}}^{N}\alpha_{t-d}(i)\,a_{ij}p_j(d)\beta_t(j) \cdot \sum_{s=t-d+1}^{t} w_{jg}\mathcal{N}\left(\boldsymbol{o}_s\mid\boldsymbol{\mu}_{jg},\boldsymbol{\Sigma}_{jg}\right) \cdot \prod_{\substack{k=t-d+1,\\ k\neq s}}^{t} b_j(\boldsymbol{o}_k)\,\boldsymbol{o}_s, \tag{11}$$

$$\eta_t^d(j,g) = \frac{1}{P(\boldsymbol{o}\mid\Lambda)}\sum_{i=1}^{N}\alpha_{t-d}(i)\,a_{ij}p_j(d)\beta_t(j) \cdot \sum_{s=t-d+1}^{t} w_{jg}\mathcal{N}\left(\boldsymbol{o}_s\mid\boldsymbol{\mu}_{jg},\boldsymbol{\Sigma}_{jg}\right) \cdot \prod_{\substack{k=t-d+1,\\ k\neq s}}^{t} b_j(\boldsymbol{o}_k)\left[\boldsymbol{o}_s-\mu_{jg}\right]\left[\boldsymbol{o}_s-\mu_{jg}\right]^{\top}. \tag{12}$$

Let us assume that the state duration probability distribution of the j-th state of an HSMM Λ is modeled by a Gaussian distribution with mean ξ_j and variance σ_j^2. The re-estimation formulas of ξ_j and σ_j^2 are derived as follows:

$$p_j\left(d_j\right) = \mathcal{N}\left(d_j\mid\xi_j,\sigma_j^2\right), \tag{13}$$

$$\overline{\xi}_j = \frac{\sum_{t_0=1}^{T}\sum_{t_1=t_0}^{T}\chi_{t_0,t_1}(j)\cdot(t_1-t_0+1)}{\sum_{t_0=1}^{T}\sum_{t_1=t_0}^{T}\chi_{t_0,t_1}(j)}, \tag{14}$$

$$\overline{\sigma}_j^2 = \frac{\sum_{t_0=1}^{T}\sum_{t_1=t_0}^{T}\chi_{t_0,t_1}(j)\cdot(t_1-t_0+1)^2}{\sum_{t_0=1}^{T}\sum_{t_1=t_0}^{T}\chi_{t_0,t_1}(j)} - \left(\overline{\xi}_j\right)^2, \tag{15}$$

where $\chi_{t_0,t_1}(j)$ is the probability of occupying the j-th state of the HSMM Λ from time t_0 to t_1, which can be written as

$$\chi_{t_0,t_1}(j) = \frac{1}{P(\boldsymbol{o}\mid\Lambda)}\sum_{\substack{i=1\\ i\neq j}}^{N}\alpha_{t_0-1}(i)\,a_{ij} \cdot \prod_{s=t_0}^{t_1} b_j(\boldsymbol{o}_s)\cdot p_j(t_1-t_0+1)\cdot\beta_{t_1}(j). \tag{16}$$

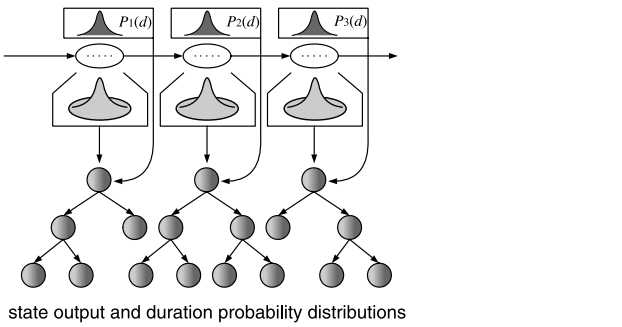
2.2 Context-Dependent Duration Modeling

There are a number of contextual factors that affect speech parameters. In HMM-based speech recognition systems, context-dependent models such as triphones have been used. However, if context-dependent models are used, the number of possible models increases exponentially. To avoid this problem, a variety of parameter sharing techniques have been developed [11]. The use of phonetic decision trees is one good solution to this problem [10].

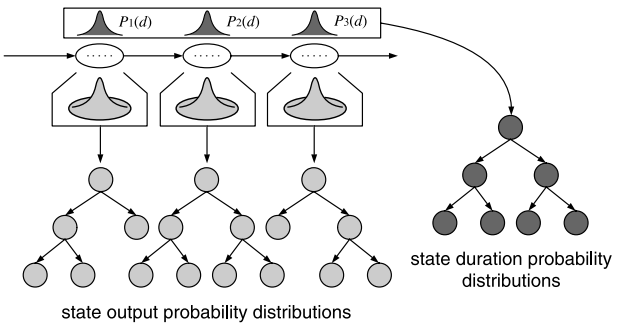
In the conventional HSMM-based speech recognition systems, either the context-independent duration model or the same parameter tying structure as of state output probability distributions was used [7] (Fig. 3 (a)). However, it is generally thought that state output and duration probability distributions have different context-dependencies. We adopted a context-dependent duration modeling technique used in HMM-based speech synthesis [9]. The state duration probabilities of each HSMM are modeled by single multi-variate Gaussian distributions whose dimensionality is equal to the number of states of the HSMM. Thus, the Gaussian distribution of the i-th dimension has the mean and variance of the state duration probability distribution for the i-th state of the HSMM. In the proposed system, state output and duration probability distributions are clustered independently by phonetic decision trees [10] (Fig. 3 (b)). Constructed phonetic decision trees represent different context-dependencies for state duration and spectral features.

2.3 HSMM-Native WFST Decoder

Most conventional HSMM-based speech recognition systems have not used state duration models in their decoders [8]. Usually, the N-best hypotheses generated by the HMMs are rescored using the HSMM likelihood. We



(a) Context-clustering that uses the same parameter tying structure as of state output probability distributions.



(b) Context-clustering that separates state duration probability distributions from HMM parameters.

Fig. 3 Context-clustering for state output and duration probability distributions.

constructed an HSMM-based speech recognition system using weighted finite-state transducers (WFSTs) to incorporate state duration models into the decoding process.

Finite-state machines have been used in many areas of computational linguistics. These transducers appear as very interesting in speech processing. WFSTs associate weights, such as probabilities, duration, penalties, or any other quantity that accumulates linearly along paths to each pair of input and output symbol sequences. This offers a unified framework representing various models used in speech and language processing [12], [13]. An integrated WFST for speech recognition can be represented as

$$H \circ C \circ L \circ G, \tag{17}$$

where H, C, L, and G are WFSTs for a state transitions network, a context-dependent model mapping, a pronunciation lexicon, and a language model, respectively.

The advantages of using WFSTs for speech decoding are

- Individually designed components can be combined.
- Each component can be individually optimized.
- The decoder is easily managed, because the network and the decoder itself are constructed individually.

Furthermore, each component can be replaced easily (Fig. 4). Using these advantages, we can easily design a speech decoder for HSMMs and fairly compare the performance of different acoustic models based on a common

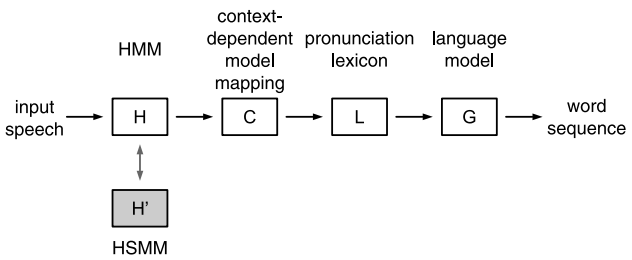


Fig. 4 WFSTs for speech recognition.

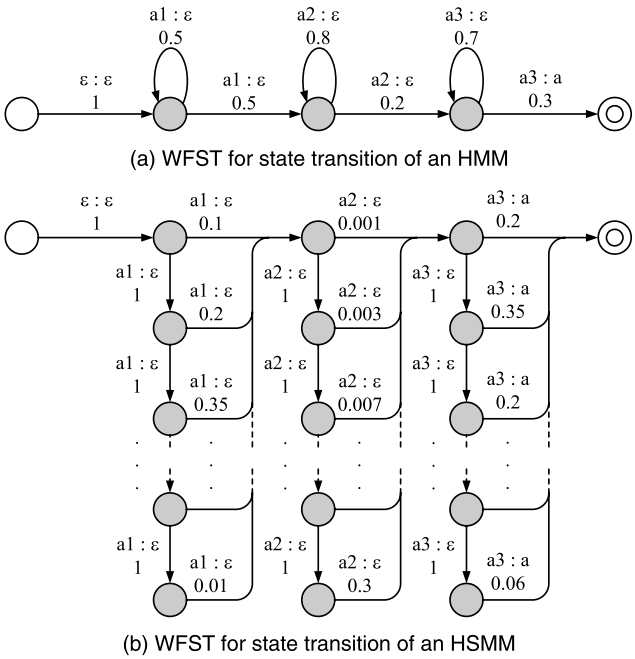


Fig. 5 WFSTs for state transitions of an HMM and an HSMM.

WFST decoding software.

Figure 5 shows the state transition of an HMM and an HSMM represented by WFSTs. All arcs of Fig. 5 (a), and Fig. 5 (b) are weighted by state transition probabilities, and state duration probabilities. The maximum state duration in Fig. 5 (b) is limited because we cannot represent infinite duration in the WFST framework. In this paper, the normalization to satisfy the probability constraint $\sum_d p_j(d) = 1$ is not applied to state duration models because no major difference was found in speech recognition performances for normalized and unnormalized state duration probabilities, respectively. It is noted that the normalization has similar effect to the duration weighting.

3. Experiments

To evaluate the performance of the proposed HSMM-based speech recognition system, speaker-dependent continuous phoneme recognition experiments were conducted on the ATR Japanese speech database B set (phonetically-balanced sentences). In all experiments, the speech data was down-

sampled from 20 kHz to 16 kHz, windowed at a 5 ms frame rate using a 25 ms Blackman window, and parameterized into 25 mel-cepstral coefficients with a mel-cepstral analysis technique. Static coefficients including the zero-th coefficients and their first and second derivatives were used as feature parameters. 3-state left-to-right structures were used and 118 questions about left and right phonetic contexts were prepared for decision tree construction. Each state output distribution was modeled by a Gaussian distribution with a diagonal covariance matrix. A WFST for decoding was constructed from WFSTs representing chained triphone HMMs and a phoneme network (phoneme-pair grammar) based on the WFST composition and determinization. Maximum and minimum state duration of each HSMM state was limited to $\xi_i \pm \sqrt{\sigma_i^2} \times 2$.

3.1 Model Size

Phonetic decision-tree-based context-clustering [10] was applied independently to state output and duration probability distributions. The MDL criterion was used to stop tree growth [14]. We changed the weight for the penalty term of c (Eq. (9) in [14]) to construct acoustic models with various numbers of parameters. The same weight c was used to cluster both state output and duration probability distributions. Thus, the number of state duration probability distributions changed according to the number of state output probability distributions.

Our WFST decoder for phoneme recognition can be represented as

$$\log(\boldsymbol{O} \mid \Lambda) \simeq \max_{\boldsymbol{Q}} \{\log P(\boldsymbol{O} \mid \boldsymbol{Q}, \Lambda) + w \log P(\boldsymbol{Q} \mid \Lambda)\} \quad (18)$$

where, $\boldsymbol{O}$, Λ, $\boldsymbol{Q}$, and w are an observation vector sequence, a sequence of phoneme HSMMs, an state sequence, and duration weight. In this experiment, we set $w = 1$ at all frames.

In the first experiment, phonetically balanced 450 sentences uttered by a speaker MHT were used for training HMMs and HSMMs. The remaining 53 sentences were used for evaluation. We fixed the beam width to 2000 and evaluated the effect of modeling state duration probability distributions.

Figure 6 shows the result. It can be seen from the figure that the proposed fully consistent HSMM-based system represents an improvement over the conventional HMM-based system in all settings.

3.2 Search Efficiency

The proposed HSMM-based system has a larger search space because the state transition WFST of HSMMs is more complex than that of HMMs. Therefore, we expected that its performance would depend strongly on beam width. To test this expectation, we fixed MDL weight c to 1 and examined the effective beam width of both the HMM- and HSMM-based systems. If the same beam width is used, the computational complexities of both systems are almost equal.

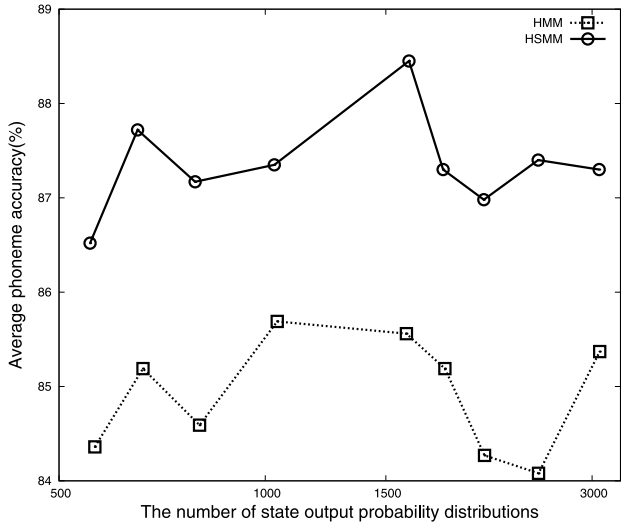


Fig. 6 Average phoneme accuracy versus the number of state output probability distributions.

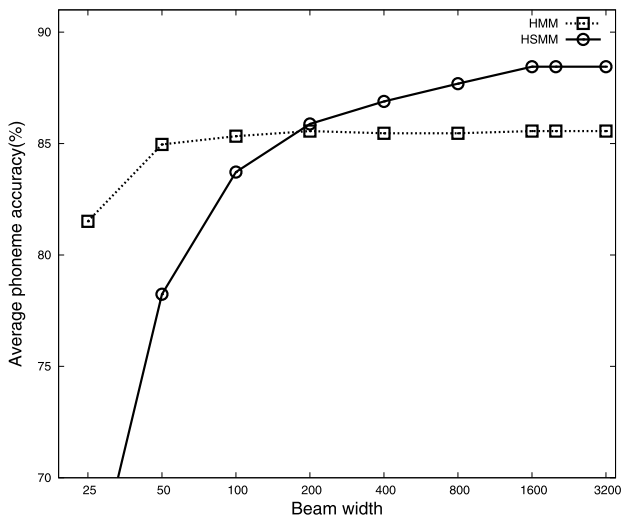


Fig. 7 Average phoneme accuracy versus beam width.

Figure 7 shows the results. It can be seen from the figure that if the beam width is lower than 200, the HSMM-based system does not perform as well as the HMM-based system. However, if the beam width is larger than 200, the HSMM-based system performs better.

3.3 Duration Weight

In the third experiment, we fixed the beam width to 2000 and evaluated the effect of duration weight.

Figure 8 shows the results. As the duration weight increased, the performance improved, peaking when the weight reached 20. At this point, performance of the HSMM-based system achieved about 48% error reduction over the HMM-based system.

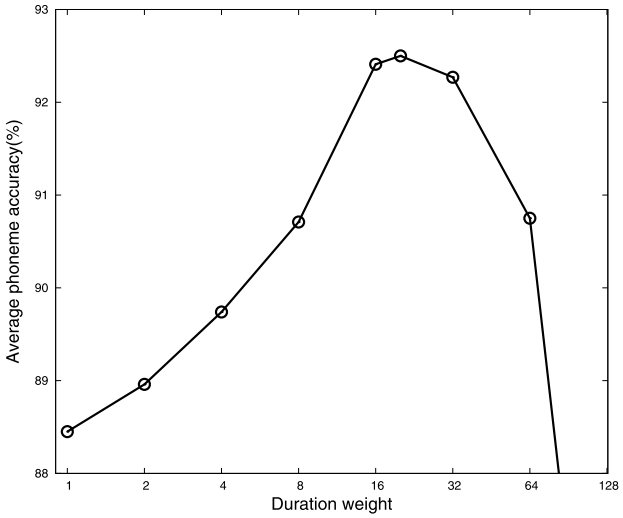


Fig. 8 Average phoneme accuracy versus duration weight.

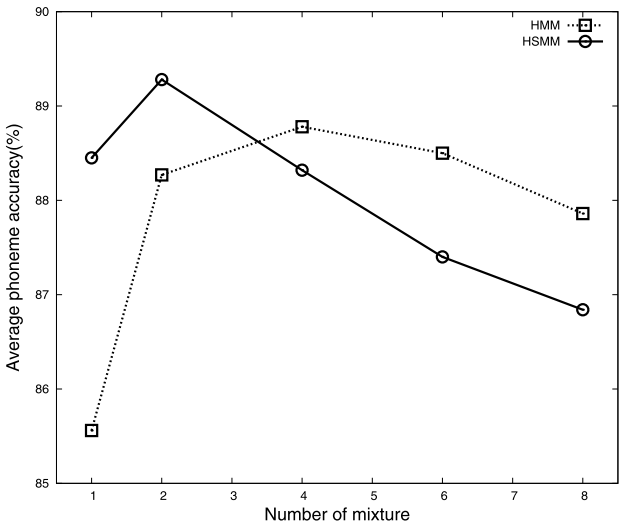


Fig. 9 Average phoneme accuracy versus number of mixture.

3.4 Number of Mixture

To test the effect of the number of mixtures, we changed the number of mixtures of both the HMM- and HSMM-based systems [†]. The same MDL weight, beam width and duration weight were used.

Figure 9 shows the results. It can be seen from the figure that if the number of mixtures is higher than 4, the HSMM-based system does not perform as well as the HMM-based system. However, comparing the best performance for each of the 2-mix HSMM-based system and the 4-mix HMM-based system, we see that the 2-mix HSMM-based system performs better.

3.5 Comparative Experiment

To investigate the effects of the three approximations mentioned in Sect. 2, we conducted a comparative experiment using 10 speakers (4 female speakers FKN, FKS, FTK, FYM, 6 male speakers MHO, MHT, MMY, MSH, MTK, MYI). In this experiment, we constructed the following 5 systems:

HMM An HMM-based system.

HSMM (train) An HSMM-based system with the approximation that state duration probability distributions were estimated based on statistics calculated by the forward-backward algorithm of the HMM [9].

HSMM (mono) An HSMM-based system with monophone state duration probability distributions.

HSMM (state) An HSMM-based system with a common state sharing structure for both state output and duration probability distributions [††]. In this experiment, context-clustering was applied using state output likelihood only.

HSMM (rescore) An HSMM-based system with the approximation that the 100 best hypotheses generated by the HMMs were rescored using the HSMM likelihood [8].

HSMM The proposed fully consistent HSMM-based system.

The beam width were fixed to 2000. Phoneme insertion penalty and duration weight were optimized for each system.

Figure 10 shows the speech recognition performance of each system. Comparing "HSMM" with "HSMM (train)" and "HSMM (rescore)", we see that the approximation in training and decoding, respectively, degrade the speech recognition performance significantly. Although difference between "HSMM" and "HSMM (mono)" is small, "HSMM" performs better than "HSMM (mono)." It is seen from Fig. 10 that "HSMM (state)" is worse than "HSMM (mono)". It seems that the lack of training data was caused by common state sharing structures of state output and duration probability distributions. By comparing "HMM" with "HSMM (mono)" and "HSMM", we found that the differences were statistically significant [16] at the 5% level. Finally, we can see that by avoiding all three approximations, the fully consistent HSMM-based system "HSMM" achieved about 9.1% error reduction over the standard HMM-based system "HMM".

[†]In this experiment, we used fast forward-backward algorithm reported in [15] to reduce computational time of mixture.

[††]It is effective in the sense of speech recognition performance that state output and duration probability distributions have independent state sharing structures. However, the system that state output and duration probability distributions have common state sharing structures performs a more effective search because their structures are combined by WFST optimization. Future work includes investigations of search efficiency with WFST optimization when state output and duration probability distributions have common state sharing structures.

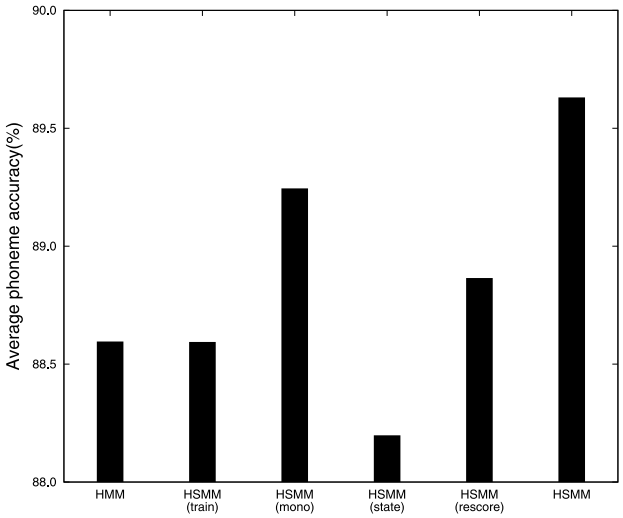


Fig. 10 Comparative experiment: Average phoneme accuracy with insertion penalty and duration weight.

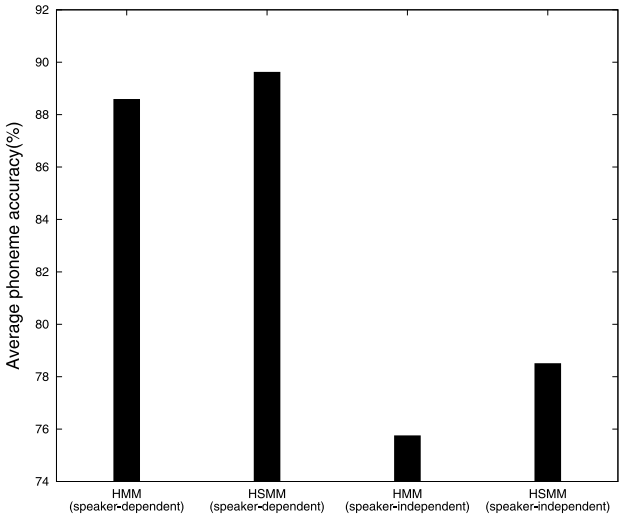


Fig. 11 Speaker-independent experiment: Average phoneme accuracy with insertion penalty and duration weight.

3.6 Speaker-Independent Experiment

To test the effects of speaker-dependency, we conducted a speaker-independent continuous phoneme recognition experiment using 10 speakers. In this experiment, nine speakers data sets and one speaker data set are used as traing and testing, respectively. The beam width were fixed to 2000. Phoneme insertion penalty and duration weight were optimized for each system.

Figure 11 shows the speech recognition performance of each system. It can be seen from the figure that the HMM-based system does not perform as well as the HSMM-based system for both speaker-dependent and speaker-independent tasks. This means that state duration modeling is effective not only for speaker-dependent tasks but speaker-independent tasks.

4. Conclusion

In this paper, we constructed a fully consistent HSMM-based speech recognition system and evaluated its performance while avoiding approximations in training, context-clustering, and decoding. The result showed an obvious improvement in phoneme recognition accuracy. Future works include evaluation on speaker independent speech recognition tasks with multi-mixture state output probability distributions and investigation other kind of distributions such as gamma, Poisson and log Gaussian distributions for state duration modeling.

References

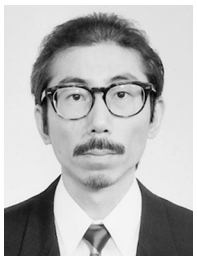
[1] M. Ostendorf, V. Digalakis, and O. Kimball, "From HMMs to segment models," IEEE Trans. Speech Audio Process., vol.4, no.5, pp.360–378, 1996.
[2] S.E. Levinson, "Continuously variable duration hidden Markov models for automatic speech recognition," Comput. Speech Lang., vol.1, pp.29–45, 1986.
[3] H. Zen, K. Tokuda, T. Masuko, and T. Kitamura, "A hidden semi-Markov model-based speech synthesis system," IEICE Trans. Inf. & Syst., vol.E90-D, no.5, pp.825–834, May 2007.
[4] J. Ferguson, "Variable duration models for speech," Proc. Symposium on the Application of Hidden Markov Models to Text and Speech, pp.143–179, 1980.
[5] Y. Ishimatsu, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Investigation of state duration model based gamma distribution for HMM-based speech synthesis," IEICE Technical Report, vol.101, no.352, pp.57–62, 2001.
[6] M.J. Russell and A.E. Cook, "Experimental evaluation of duration modeling techniques for automatic speech recognition," Proc. ICASSP1987, vol.1, pp.2376–2379, 1987.
[7] M. Wan. Koo, S.J. Park, and D.Y. Son, "Context dependent phoneme duration modeling with tree-based state tying," Proc. INTERSPEECH2004, vol.1, pp.721–724, 2004.
[8] V.R.R. Gadde, "Modeling word duration," Proc. ICSLP2000, vol.1, pp.601–604, 2000.
[9] T. Yoshimura, K. Tokuda, T. Masuko, T. Kobayashi, and T. Kitamura, "Simultaneous modeling of spectrum, pitch and duration in HMM-based speech synthesis," Proc. EUROSPEECH, vol.5, pp.2347–2350, 1999.
[10] J. Odell, The use of context in large vocabulary speech recognition, Ph.D. Thesis, Cambridge University, 1995.
[11] K.F. Lee, "Context-dependent phonetic hidden Markov models for speaker-independent continuous speech recognition," IEEE Trans. Acoust. Speech Signal Process., vol.38, no.4, pp.599–609, 1990.
[12] M. Mohri, F. Pereira, and M. Riley, "Weighted finite-state transducers in speech recognition," Proc. ASR2000, pp.97–106, 2000.
[13] C. Allauzen and M. Mohri, "Generalized optimization algorithm for speech recognition transducers," Proc. ICASSP2003, vol.1, pp.352–355, 2003.
[14] K. Shinoda and T. Watanabe, "Acoustic modeling based on the MDL criterion for speech recognition," Proc. EUROSPEECH, vol.1, pp.99–102, 1997.
[15] S.Z. Yu and H. Kobayashi, "An efficient forward-backward algorithm for an explicit-duration hidden Markov model," IEEE Signal Process. Lett., vol.10, no.1, pp.11–14, 2003.
[16] S. Nakagawa and H. Takagi, "Statistical methods for comparing pattern recognition algorithms and comments on evaluating speech recognition performance," J. Acoust. Soc. Jpn., vol.50, no.10, pp.849–854, 1994.

Keiichiro Oura was born in 1982. He receieved the B.E., and M.E. degrees in computer science, and computer science and Engineering from the Nagoya Institute of technology, Nagoya, Japan in 2003, and 2005, respectively. He is currently a Doctor's candidate at the Nagoya Institute of technology. His research interests include statistical speech recognition and synthesis. He is a student member of the Acoustical Society of Japan.

Heiga Zen was born in Osaka, Japan, on March 4, 1979. He received the A.E. degree in electronic and information engineering from Suzuka National College of Technology, Suzuka, Japan, in 1999, and the B.E., M.E., and Dr.Eng. degrees in computer science, electrical and computer engineering, and computer science and engineering from Nagoya Institute of Technology, Nagoya, Japan, in 2001, 2003, and 2006, respectively. During 2003, he was an intern researcher at ATR Spoken Language Translation Research Laboratories (ATR-SLT), Kyoto, Japan. From June 2004 to May 2005, he was an intern/co-op researcher in the Human Language Technology group at IBM T.J. Watson Research Center, Yorktown Heights, NY. He is currently a postdoctoral fellow at Nagoya Institute of Technology. His research interests include statistical speech recognition and synthesis. He received the Awaya and Itakura Awards from the Acoustical Society of Japan (ASJ) in 2006 and 2008, respectively, and the TELECOM System Technology Prize from the Telecommunications Advancement Foundation Award, Japan, in 2008. He is a member of ASJ and ISCA.

Yoshihiko Nankaku received the B.E. degree in Computer Science, and the M.E. and Dr.Eng. degrees in the Department of Electrical and Electronic Engineering from the Nagoya, Institute of Technology, Nagoya Japan, in 1999, 2001, and 2004 respectively. He is currently an Assistant Professor of Nagoya Institute of Technology. His research interests include image recognition, speech recognition and synthesis and multimodal interface. He is a member of the Acoustical Society of Japan.

Akinobu Lee was born in Kyoto, Japan, on December 19, 1972. He received the B.E. and M.E. degrees in information science, and the Ph.D. degree in informatics from Kyoto University, Kyoto, Japan, in 1996, 1998, and 2000, respectively. During 2000–2005, he was an Assistant Professor in Nara Institute of Graduate School of Information Science, Nara Institute of Science and Technology, Nara, Japan. He is now an Associate Professor of Nagoya Institute of Technology. His research interests include speech recognition and spoken language understanding. He is a member of the IEE, ISCA, IPSJ and ASJ.



Keiichi Tokuda received the B.E. degree in electrical and electronic engineering from the Nagoya Institute of Technology, Nagoya, Japan, in 1984 and the M.E. and Dr.Eng. degrees in 8 IEICE Trans. Fundamentals, vol.E86.A, no.5 May 2003 information processing from the Tokyo Institute of Technology, Tokyo, Japan, in 1986 and 1989, respectively. From 1989 to 1996, he was a Research Associate in the Department of Electronic and Electric Engineering, Tokyo Institute of Technology. From 1996 to 2004, he was an Associate Professor in the Department of Computer Science, Nagoya Institute of Technology, where he is currently a Professor. He is also an Invited Researcher at the ATR Spoken Language Communication Research Laboratories, Kyoto, Japan, and was a Visiting Researcher at Carnegie Mellon University, Pittsburgh, PA, from 2001 to 2002. His research interests include speech coding, speech synthesis and recognition, and statistical machine learning. Prof. Tokuda is a corecipient of the Paper Award and the Inose Award, both from the IEICE in 2001, and the TELECOM System Technology Prize from the Telecommunications Advancement Foundation Award, Japan, in 2001 and 2008. He was a member of the Speech Technical Committee of the IEEE Signal Processing Society. He is a member the ISCA, IPSJ, ASJ, and JSAI.