Discriminative training and explicit duration modeling for HMM-based automatic segmentation

Yi-Jian Wu a,b,*, Hisashi Kawai b, Jinfu Ni b, Ren-Hua Wang a

a IFly Speech Laboratory, University of Science and Technology of China, Hefei, Anhui 230026, China
b ATR, Spoken Language Translation Research Laboratories, Kyoto 619-0288, Japan

Received 23 July 2004; received in revised form 9 February 2005; accepted 8 March 2005

Abstract

HMM-based automatic segmentation has been popularly used for corpus construction for concatenative speech synthesis. Since the most important reasons for the inaccuracy of HMM-based automatic segmentation are the HMM training criterion and duration control, we will study these particular issues. For the HMM training, we apply the discriminative training method and introduce a new criterion, named Minimum Segmentation Error (MSGE). In this method, a loss function directly related to the segmentation error is defined, and parameter optimization is performed by the Generalized Probabilistic Descent (GPD) algorithm. For the duration control problem, we apply explicit duration models and propose a two-step-based segmentation method to solve the problem of computational cost, where the duration model is incorporated in a postprocessor procedure. From the experimental results, these two techniques significantly improve segmentation accuracy with different focuses, where the MSGE-based discriminative training focuses on improving the accuracy of sensitive boundary, i.e., a boundary where an error in segmentation is likely to cause a noticeable degradation in speech synthesis quality, and the explicit duration modeling focuses on eliminating large errors. After combining these two techniques, the error average was reduced from 6.86 ms to 5.79 ms on Japanese data, and from 8.67 ms to 6.61 ms on Chinese data. Simultaneously, the number of errors larger than 30 ms were reduced 25% and 51% on Chinese and Japanese data, respectively.

© 2005 Elsevier B.V. All rights reserved.

Keywords: Automatic segmentation; Discriminative training; Minimum segmentation error; Explicit duration modeling; Speech synthesis
1. Introduction

Recently, corpus-based concatenative speech synthesis has become popular due to its high quality, which is directly related to the accuracy of the phonetic labeling of the corpus. Since the labeling task needs a lot of human effort and a long time, especially for a large corpus (e.g., 100 h), automatic segmentation (AS) has been very important for corpus-based speech synthesis (Ljolje et al., 1997; Carvalho et al., 1998), providing consistent and accurate phonetic labeling with high efficiency. In addition to the HMM-based method adopted from automatic speech recognition (ASR) in (Ljolje et al., 1997), many methods (Sethy and Narayanan, 2002; van Santen and Sproat, 1999; Malfrere and Dutiot, 1997; Kim and Conkie, 2002) have been proposed for the AS task, and each has obtained a certain level of performance. In (van Santen and Sproat, 1999), the broad/narrow band edge detectors are applied to each diphone or diphone class for boundary detection. Also, (Sethy and Narayanan, 2002) introduced boundary models, which are trained from existing speech corpora, and used as a postprocessor to refine the boundary placements generated by the HMM-based method. In (Malfrere and Dutiot, 1997), a high-quality speech synthesizer is used to produce a synthetic reference signal, and the original speech signal is then temporally aligned on this reference by using the dynamic time warping (DTW) algorithm.

As a by-product of ASR, the HMM-based segmentation method has been popularly used. Although the current results are quite impressive, there are also shortcomings that prevent it from achieving even better performance. Since the most important reasons for the inaccuracy of HMM-based automatic segmentation are the HMM training criterion and duration control, we will study these particular issues here.

The conventional HMM training method is adopted from speech recognition (Rabiner, 1989), which is based on Maximum Likelihood Estimation (MLE) criteria (via a powerful training algorithm, the Expectation Maximization algorithm). In other words, this training method links the segmentation task to the problem of distribution estimation, and the HMMs are built to identify the phonetic segments, not to detect the boundary between the phonetic segments. For example, to train a phoneme HMM based on MLE criteria, the method may use the last 5–10 frames of each training data related to this phoneme to train the last state model. However, for segmentation, only the last 1–2 frames near the boundary are actually suitable for training the last state model. It is also the same for the first state model. This kind of inconsistency between the training and the application of HMM limits the segmentation performance.

In recent years, the discriminative training method and the criteria of Minimum Classification Error (MCE) based on the Generalized Probabilistic Descent (GPD) framework has been successful in training HMM for speech recognition (Juang et al., 1997; McDermott, 1997; Chou, 2000), and, to a certain extent, segmentation can be regarded as a state recognition task with known transcription. This prompts us to apply the discriminative training method and the corresponding criteria for the segmentation task. In this paper, a new criteria, called Minimum SeGmentation Error (MSGE), is proposed to train the HMM under the GPD framework. In this method, we defined a loss function directly related to segmentation errors. By minimizing the overall empirical loss under the GPD framework, the segmentation errors could also be minimized.

Another issue is related to the inappropriate duration model, which is implied by the transition probability, assuming the duration probability distribution of each state as being a geometric distribution. In practice, the exponential distribution is usually inappropriate to the actual duration distribution (Levinson, 1986). Due to the absence of an explicit duration model, some large errors with implausible duration are inevitable in HMM-based automatic segmentation. Several approaches (Burshtein, 1996; Russell and Moore, 1985; Yoma et al., 2001) to duration modeling have been proposed for speech recognition and have improved the performance to a certain extent. Due to the difference between the speech recognition task and the AS task, we should find appropriate duration modeling for segmentation.
This paper is organized as follows. In Section 2, we first briefly review the GPD algorithm for parameter optimization. The MSGE-based HMM training procedure including a loss function definition and a parameter updating schedule are then presented in detail. In Section 3, we define a phone-based duration model and introduce a two-step-based segmentation method to solve the problem of computational cost, where the duration model is incorporated in a postprocessor procedure. Weight optimization for the duration model based on a hill-climbing algorithm is also presented. In Section 4, experiments are performed to evaluate the effect of MSGE-based discriminative training and explicit duration modeling on the segmentation. Finally, we summarize the paper in Section 5.

2. MSGE-based HMM training algorithm

Here, the discriminative training method is adopted and a new criterion, named Minimum Segmentation Error (MSGE), is introduced to train the HMMs under the GPD framework. In this method, we define a loss function based on a new measurement of segmentation error. The HMM parameters are optimized under the GPD framework by minimizing the overall empirical loss.

2.1. Generalized probabilistic descent

First, we give a brief introduction of the GPD algorithm. For a given loss function \( \ell(X, A) \), where \( X \) is a feature vector and \( A \) represents the system parameters, we want to optimize \( A \) to minimize the overall expectation loss:

\[
L(A) = E[\ell(X, A)] = \int \ell(X, A)p(X)dX,
\]

where \( p(X) \) is a priori distribution. Since we do not know the a priori distribution, we cannot evaluate the expected loss directly. The Generalized Probabilistic Descent (GPD) algorithm (Blum, 1954) is a very powerful algorithm that can be used to accomplish this task. In a GPD framework, the target loss function is minimized according to an iterative procedure

\[
A_{t+1} = A_t - \varepsilon_t U_t \nabla \ell(X_t, A)|_{A=A_t},
\]

where \( U_t \) is a positive definite matrix, \( X_t \) is the \( t \)th training sample used in the sequential training process, and \( \varepsilon_t \) is a sequence of positive numbers, that satisfies the following:

\[
\sum_{i=1}^{\infty} \varepsilon_i \to \infty
\]

and

\[
\sum_{i=1}^{\infty} \varepsilon_i^2 < \infty.
\]

In the above, an infinite number of training samples are required for convergence. In practice, only a finite number of samples are available. However, we can minimize the empirical loss:

\[
L_0(A) = \frac{1}{N} \sum_{i=1}^{N} \ell(X_i, A) = \int \ell(X, A)p_N(X)dX
\]

under the GPD framework. With sufficient training samples, the empirical loss converges to the actual expected loss. It should be noted that the GPD framework is a general framework for various definitions of the loss function. A more detailed introduction and discussion of the GPD algorithm can be found in the literature (Blum, 1954; Chou, 2000).

2.2. Measurement of segmentation error

The conventional measurement of segmentation error is usually defined as the time difference in boundary location between human labeling and automatic labeling, i.e., error length. According to this definition, the segmentation errors are discrete (in frame scale) and not explicitly related to the parameters of the HMM. Therefore, gradient-based optimization methods cannot be used to minimize the segmentation errors directly. It is necessary to find an appropriate measurement of segmentation error.

HMM-based segmentation is a state alignment procedure usually performed by a Dynamic Programming algorithm (e.g., Viterbi). For simplification, we look into the segmentation procedure of a sample \( X \) which consists of two connected segment...
units $X_1$ and $X_2$, i.e., $X = \{X_1, X_2\}$. In the DP algorithm, the likelihood of the best state alignment is calculated by
\[
g_b(X; A) = \max_Q g(X, Q; A) = g(X, \overline{Q}_b; A),
\]
where $\overline{Q}_b$ is the optimal state sequence with maximum likelihood, which is calculated as
\[
g(X, \overline{Q}_b; A) = \log P(X, \overline{Q}_b; A)
\]
\[= \sum_{i=1}^{T} \left[ \log a_{\bar{q}_i, \bar{q}_i} + \log b_{\bar{q}_i}(x_i) \right] + \log \pi_{\bar{q}_0},
\]
where $p_{\bar{q}_0}$ is initial state probability, $a_{\bar{q}_i, \bar{q}_i}$ is transition probability, and $b_{\bar{q}_i}(x_i)$ is output probability distribution.

With the optimal state alignment, the corresponding phonetic boundary is labeled at time $t'$, which satisfies the conditions that $\bar{q}_{t'-1}$ is the final state of the first unit and $\bar{q}_{t'}$ is the first state of the next unit. If this boundary is not the same as the hand-labeled boundary, i.e., the correct boundary, the optimal state alignment is regarded as an “incorrect” state alignment. Also, the “correct” state alignment is defined as the optimal state alignment with the correct phonetic boundary restriction, which satisfies the following:
\[
g_c(X; A) = g_1(X_1, \overline{Q}_1; A) + g_2(X_2, \overline{Q}_2; A)
\]
\[= g(X, \overline{Q}_c; A),
\]
where $\overline{Q}_1$ and $\overline{Q}_2$ are respectively the optimal state sequences of $X_1$ and $X_2$, and $\overline{Q}_c = \{\overline{Q}_1, \overline{Q}_2\}$.

Accordingly, we defined error degree as the likelihood difference between the incorrect and the correct state sequence, i.e.,
\[
E_d = g_b(X, A) - g_c(X, A),
\]
where $g_b(X, A)$ and $g_c(X, A)$ are the likelihood of the incorrect and correct state sequences respectively. When the segmentation is correct, i.e., $\overline{Q}_b = \overline{Q}_c$, $E_d$ is equal to 0. If $E_d$ is larger than 0, this indicates that the segmentation is incorrect and the value of $E_d$ reflects how large the segmentation error is in a certain aspect. In order to find out the meaning of the error degree in depth, we analyzed the correlation between error degree and error length.

In the preliminary experiment, HMMs trained by MLE criteria were used to segment the Japanese training data (the details of the data information can be found in Section 4.1). The correlation between error degree and error length was then analyzed from all segmentation errors, and the results of some typical boundaries are shown in Fig. 1. As seen in the figure, error degree is nearly linear with error length, and for different boundary types, the slope is different. That means the linear correlation is context dependent. For the sensitive boundary, e.g., the boundary between plosive and vowel, or fricative and vowel, etc., the slope is relatively large, which means error degree is sensitive to error length. For the insensitive boundary between vowel and vowel, or nasal and vowel, the slope is relatively small, i.e., error degree is less sensitive to error length. This characteristic is identical with the requirement of concatenative speech synthesis, which is quite sensitive to the segmentation accuracy of plosive segments as a plosive segment with an imprecise boundary might result in two bursts or no burst in synthetic speech, and less sensitive to the accuracy of vowel segments. From this point of view, error degree is a more exact factor than error length to measure the segmentation error. Because of the correlation between error degree and error length, the minimization of error degree is also related to the minimization of error length.

![Fig. 1. Correlation between error degree and error length.](image-url)
From the figure, it should be noted that the slope is small for the boundary between vowel and plosive. In fact, we regard this boundary type as the insensitive boundary, because the transition between the vowel and the closure of the plosive is smoothed. From the waveform, we can see that the tail of the vowel has very low energy, which is analogous with the closure of the plosive.

2.3. Loss function definition

Based on the new measurement of segmentation error, we defined the loss function as

$$\ell(A) = E^2_tE_d = E^2_t(g_b(X, A) - g_c(X, A)), \quad (10)$$

where $E_t$ is the error length and $x$ is a positive number. In this loss function, $E^2_t$ is regarded as a constant number in the optimization procedure by the GPD algorithm, so the loss function is differentiable with respect to the parameter. The meaning of $E^2_t$ can be explained as follows.

On the one hand, it indicates the consideration of the error length of the training sample, and $x$ is the weight of the error length. When $x$ is larger than 0, it means we have more focus on a training sample with larger errors, accordingly updating the parameters in a larger scale. Thus, by adding $E^2_t$, it provides a flexible way to optimize the parameters for different focuses. On the other hand, $E^2_t$ means the weight of the training sample, i.e., the same performance can be achieved by repeating the training sample $E^2_t$ times when the loss function is defined as $E_d$ only.

Combined with $E_t$ and $E_d$, this definition of loss function is very meaningful, reflecting both the explicit error length and the inherent error degree. Moreover, under this definition, the loss function is continuous, differentiable, and directly related to the HMM parameters. By using the gradient-based optimization method (e.g., GPD), the loss function can be minimized, which relates to a minimization of the segmentation error.

2.4. Parameter updating

Now we optimize the parameters under this loss function by the GPD algorithm. For a state $j$ of HMM $h$ with $M$ mixtures, the output probability distribution is

$$b_{h,j}(x_t) = \sum_{m=1}^{M} c_{h,j,m} b_{h,j,m}(x_t)$$

$$= \sum_{m=1}^{M} c_{h,j,m} G[x_t; \mu_{h,j,m}, R_{h,j,m}], \quad (11)$$

where $b_{h,j,m}(\cdot)$ is the output probability of one mixture, $G[\cdot]$ is the normal Gaussian distribution, and $c_{h,j,m}$, $\mu_{h,j,m}$, $R_{h,j,m}$ are mixture weights, mean vector, and covariance matrix, respectively.

It should be noted that the HMM as a probability measure has some original constraints, such as: (1) the function being positive, (2) $\sum_{m} c_{h,j,m} = 1$ for all $h, j$, and (3) $\sigma_{h,j,m,l} > 0$. In order to maintain these constraints during parameter adaptation, we should take some parameter transformations as follows:

$$c_{h,j,m} \rightarrow \tilde{c}_{h,j,m}, \quad \text{where}$$

$$c_{h,j,m} = \frac{\exp(\tilde{c}_{h,j,m})}{\sum_{m} \exp(\tilde{c}_{h,j,m})}, \quad (12)$$

$$\mu_{h,j,m} \rightarrow \tilde{\mu}_{h,j,m} = \mu_{h,j,m} R_{h,j,m}^{-1}, \quad (13)$$

$$R_{h,j,m} \rightarrow \tilde{R}_{h,j,m} = \log(R_{h,j,m}). \quad (14)$$

The transformation in (13) is important to design the step size for convergence. More discussion about the parameter transformation can be found in (Chou, 2000).

For a sample $X_t$ in the training set, the adaptation of parameters is as follows:

$$A_{h,j,m}(n+1) = A_{h,j,m}(n) - e \frac{\partial \ell(X_t; A)}{\partial A_{h,j,m}}|_{A=A_{h,j,m}}, \quad (15)$$

where

$$\frac{\partial \ell(X; A)}{\partial A_{h,j,m}} = E^2_t \frac{\partial (g_b(X, A) - g_c(X, A))}{\partial A_{h,j,m}}$$

$$= E^2_t \sum_{t=1}^{T} (\delta(q_{bt} - j) - \delta(q_{ct} - j)) \times b_{h,j}(x_t) \frac{\partial b_{h,j}(x_t)}{\partial A_{h,j,m}}, \quad (16)$$
where \( \delta(\cdot) \) denotes the Kronecker delta function. For the mean vector, the updating rule is
\[
\frac{\partial b_{h,j}(x_i)}{\partial \mu_{h,j,m}} = c_{h,j,m}b_{h,j,m}(x_i)R^{-1}_{h,j,m}(x_i - \mu_{h,j,m}).
\] (17)

Finally,
\[
\mu_{h,j,m}(n + 1) = \mu_{h,j,m}(n + 1)R_{h,j,m}.
\] (18)

Similarly, for the covariance matrix \( R_{h,j,m} \), the updating rule is
\[
\frac{\partial R_{h,j,m}}{\partial \mu_{h,j,m}} = c_{h,j,m}b_{h,j,m}(x_i)
\]
\[
\times \left( (R^{-1}_{h,j,m}R^{-1}_{h,j,m}(x_i - \mu_{h,j,m})(x_i - \mu_{h,j,m})^T - I_D) \right).
\] (19)

where \( I_m \) is a identity matrix. Finally,
\[
R_{h,j,m}(n + 1) = \exp\{R_{h,j,m}(n + 1)\}. \] (20)

Also, the mixture weight update is
\[
\frac{\partial b_{h,j}(x_i)}{\partial c_{h,j,m}} = b_{h,j,m}(x_i)c_{h,j,m}(1 - c_{h,j,m}).
\] (21)

Finally,
\[
c_{h,j,m}(n + 1) = \frac{\exp(c_{h,j,m}(n + 1))}{\sum_i \exp(c_{h,j,m}(n + 1))}.
\] (22)

The meaning of the updating rule can be explained as follows. In Eq. (16), \( \delta(q_{bt} - j) - \delta(q_{ct} - j) \) can be rewritten as
\[
\delta(q_{bt} - j) - \delta(q_{ct} - j) = \begin{cases} 
0, & q_{bt} = q_{ct}, \\
1, & q_{bt} \neq q_{ct}, j = q_{bt}, \\
-1, & q_{bt} \neq q_{ct}, j = q_{ct}.
\end{cases}
\] (23)

Let us look at the case of \( \delta(q_{bt} - j) - \delta(q_{ct} - j) = -1 \), i.e., \( q_{bt} \neq q_{ct} \) and \( j = q_{ct} \). From Eq. (17), without loss of generality, we assume \( x_i - \mu_{h,j,m} > 0 \), then \( \frac{\partial b_{h,j}(x_i)}{\partial \mu_{h,j,m}} > 0 \). So, we can see the second component \( -\frac{\partial b_{h,j}(x_i)}{\partial \mu_{h,j,m}} \) of the right part of Eq. (15) is larger than zero, which means that the mean parameter \( \mu_{h,j,m}(n + 1) \) becomes larger, i.e., close to the input vector \( x_i \). Also, the inverse phenomena can be found in the case of \( \delta(q_{bt} - j) - \delta(q_{ct} - j) = 1 \), i.e., \( q_{bt} \neq q_{ct} \) and \( j = q_{bt} \). From this point of view, when the best state alignment differs from the correct state alignment, the updating rule is to move the parameters of the correct state model “close to” the input vector and move the parameters of incorrect state model “far away” from the input vector.

3. Explicit duration modeling

3.1. Definition of duration model

In the conventional HMM, the duration model is implied by the transition probability, assuming the duration probability distribution of each state as being a geometric distribution, i.e., \( p_i(\tau) = a_i^{-1}(1 - a_i) \), where \( i \) is the state, \( p_i(\tau) \) is the state duration, and \( a_i \) is the state self transition probability. In practice, the exponential distribution is usually inappropriate to the actual duration distribution (Russell and Moore, 1985). Due to this, a large error with abnormal duration can occur in HMM-based automatic segmentation. In order to solve this problem, it is necessary to incorporate an appropriate duration model. Different methods for duration modeling have been proposed for speech recognition (Russell and Moore, 1985; Yoma et al., 2001). As these methods are not designed for the segmentation task, where duration modeling is applied to the temporal constraint and path pruning without considering the statistical property of the duration for each phoneme, we should apply an appropriate duration model for the segmentation task.

Let us take a look at the segmentation procedure of an input vector sequence \( X \) with \( N \) segments. For a state alignment \( q = \{q_1, q_2, \ldots, q_N\} \), where each state sequence \( q_i \) corresponds to one segment, the vector sequence is accordingly divided into \( X = (X_1, X_2, \ldots, X_N) \). Without an explicit duration model, the conventional likelihood is calculated by
\[
L_{\text{con}}(X, q; A) = \sum_{i=1}^{N} \log(P_o^i(X_i, q_i; A_i)),
\] (24)

where \( P_o^i(X_i, q_i; A_i) \) is the output probability of the acoustic model, in which the transition probability is included. Combining the acoustic model with
the explicit duration model, the new likelihood is calculated by

$$L_{\text{EDM}}(X, q; A) = \sum_{i=1}^{N} (\log(P^a_i(X_i, q_i; A_a)) + w^d_i \log(P^d_i(T_i; A_d))),$$

(25)

where $P^a_i(T_i; A_a)$ and $w^d_i$ are the output probability of the duration model and the weights of the duration model, and $T_i$ is the length of $q_i$, i.e., the length of each segment. Under the new definition of likelihood calculation, the segmentation procedure is used to find the optimal state alignment, which satisfies the following:

$$\tilde{q} = \arg\max_q L_{\text{EDM}}(X, q; A)$$

$$= \arg\max_{q, T_i} \sum_{i=1}^{N} (\log(P^a_i(X_i, q_i; A_a)) + w^d_i \log(P^d_i(T_i; A_d))),$$

(26)

where $\tilde{q}$ is the best state sequence.

For explicit duration modeling, one important thing is to choose the carrier, which might be a state, phone, syllable, etc. In order to directly relate to the aim of segmentation, the carrier of the duration model is defined as the basic segment, i.e., if the basic segment is a phone, the duration model is a phone-based duration model, etc. In phonetic segmentation, the duration can be regarded as either discrete variance in frame scale or continuous variance in time scale. Accordingly, the probability distribution of the duration model can be a Gamma or Gaussian distribution (Russell and Moore, 1985). In order to maintain the consistency of the likelihood calculation between the duration model and the acoustic model, we adopt the Gaussian distribution here, i.e.,

$$P^d(T; A_d) = \frac{1}{\sqrt{2\pi\nu^d}} \exp \left(-\frac{1}{2} \left(\frac{T - m^d}{\nu^d}\right)^2\right),$$

(27)

where $m^d$ and $\nu^d$ are the mean and the variance of the duration model, respectively.

### 3.2. Two-step-based segmentation method

One reason that the explicit duration model was not adopted in the conventional HMM is the problem of computational complexity. With this explicit duration modeling, the DP algorithm cannot be applied. If we directly search for the optimal state alignment based on Eq. (3), the computational cost is excessively large, i.e., on the order of $M^N$, where $N$ and $M$ are the number of segments and frames, respectively. To reduce the computational cost to an acceptable degree, we introduce a two-step-based method to perform the segmentation with the explicit duration model, where the duration model is incorporated in a postprocessor procedure, which is performed as follows.

1. **First step.** In the first step, the explicit duration model is not taken into account in the segmentation. Without the explicit duration model, the DP algorithm can be performed with high efficiency, which is similar to the conventional method.

2. **Second step.** Combining the acoustic model with the explicit duration models, we use a heuristic technique to search for the optimal path based on the result of the first step. This is performed in an iterative procedure as shown in Fig. 2.

Please notice that the new likelihood is calculated only with the segments inside the window centered at the boundary, which means that the boundary location is optimized locally in each step. After several iterations of the local optimization,
the result will be convergent and close to the global optimum. The preliminary experiments showed that the optimization results with different window sizes have a slight difference. However, they also showed that the difference could be compensated by the appropriate weights. For high efficiency, the window size adopted in the later experiments is fixed to 2.

Compared to the conventional computational cost of segmentation, the extra cost of this method is the cost of the second step. In the second step, the cost of one loop is comparable to the conventional cost. Commonly, it takes about 2–5 iterations to converge to the optimum result. Therefore, the total computational cost of this method is just several times that of the conventional cost, which is acceptable for segmentation as it is an off-line task.

3.3. Effect of explicit duration modeling

In our preliminary experiments, we investigated the effect of explicit duration modeling on segmentation by applying different weight coefficients. For simplification, the weights of the duration model for each phoneme are set to the same coefficients. The mean and variance of the duration model for each phoneme are initialized with the statistical parameters calculated from the training data. The experiments were performed both on Chinese and Japanese data to examine the language dependency of the effect. It should be noted that the HMM without the duration model, i.e., with the acoustic model only, is regarded as the baseline model. Detailed data and modeling information can be found in Section 4.1.

3.3.1. Effect on Chinese

The effect of the explicit duration modeling on Chinese data is shown with different aspects in Fig. 3. Please note that the values in the figure are the differences in segmentation accuracy between the current model and the baseline model. After combining the acoustic model with the duration model, the average lengths of the best 80–100% segmentation errors drop when the weight is increased from 0 to 10, and become large when the weight is increased from 10 to 50 in Fig. 3(a). The average lengths of the worst 5% and 10% segmentation errors are shown in Fig. 3(b). As this figure shows, the tendency of the improvement is similar to that shown in Fig. 3(b), except that the average error still drops when the weight is increased from 10 to 20. Another difference is that the improvement in Fig. 3(b) is much larger than that in Fig. 3(a). As we expected, these results show that the duration model has more of an effect on eliminating large segmentation errors.

![Fig. 3. Segmentation accuracy on Chinese data: (a) the error average of the best (80, 85, 90, 95, 100) percent of boundary; (b) the error average of the worst (5, 10) percent of boundary.](image-url)
3.3.2. Effect on Japanese

The effect of the duration model on Japanese data is shown in Fig. 4, which is similar to that on Chinese data. Also the duration model has more of an effect on improving large errors. The average of the worst 5% segmentation errors drops nearly 10 ms when the weight is set to 20. The only difference is that the effect of the duration model on Japanese is much larger than that on Chinese, which may be caused by the fact that the phoneme duration is more stable in Japanese than in Chinese. For example, if the phoneme duration is constant in the extreme case, the duration model has the most effect, i.e., even without the acoustic model, the duration model is sufficient for the segmentation task.

From the results both on Chinese and Japanese data, it can be seen that the accuracy of segmentation was improved after incorporating the explicit duration models with appropriate weights. And the effect of the duration model is language dependent. If the phoneme duration in the language is very stable, the duration model has a large effect, and vise versa. Nevertheless, the duration model has a considerable effect on improving segmentation errors, especially on eliminating large errors.

3.4. Weight optimization

The preliminary experiment showed that the effect of the duration model critically depends on the aptness of the weight of the duration model, and it was varied for different phonemes. Hence, it is very important to optimize the weight for each phoneme. As the duration model is incorporated as a postprocessor and segmentation is based on local optimization, the gradient-based optimization methods are not suitable for this task. Here, we adopted a hill-climbing algorithm to optimize the weight coefficients of the duration model. It was performed as follows:

(a) For each phoneme, the weight of the duration model was initialized and the segmentation accuracy was evaluated.
(b) Try to increase (decrease) the weight with certain step size, and re-evaluate the segmentation accuracy.
(c) If the segmentation accuracy was improved, go back to (b). Otherwise the procedure finished.
(d) Perform the above optimization procedure for each phoneme in turn.

From the experiments, the weights of the duration model converged after 2–3 iterations of the above procedure. In order to improve the efficiency of the optimization procedure, the step size set to a big value (e.g., 3–5) in the first iteration, and a small value (e.g., 0.5–1) for the second and third iteration. Actually, the weights can be

![Fig. 4. Segmentation accuracy on Japanese data: (a) the error average of the best (80, 85, 90, 95) percent of boundary; (b) the error average of the worst (5, 10) percent of boundary.](image-url)
optimized more if the step size set to smaller value. But the improvement of segmentation accuracy is very limited, which can be ignored.

4. Evaluation experiments

In this section, experiments were carried out to evaluate the effect of MSGE-based HMM training and explicit duration modeling on segmentation.

4.1. Experiment information

We performed experiments on both Chinese and Japanese corpora, which are based on a single speaker respectively. Detailed information of the corpora and the HMM modeling is shown below.

(1) Chinese: The training and testing data consisted of 1000 and 680 sentences, including 27,312 and 15,872 phones respectively, and all the data had been hand-labeled. The phone set had 60 phonemes, including 21 initials, 37 finals, pause and silence. Monophone HMMs were adopted and the number of states was three for initials, pause and silence, and five for finals, and the number of mixtures was set to five for each phoneme. The acoustic feature was 16-order MFCC and energy, and the delta coefficients. The analysis window size and shift were 20 ms and 5 ms respectively.

(2) Japanese: The training and testing data consisted of 2263 and 501 phonetically balanced sentences, including 185,404 and 30,706 phones respectively. All the data were uttered by the same speaker. The phone set used here included 60 phonemes. Also, monophone HMMs were used, and the number of states and mixtures were respectively three and five for each phoneme. The parameters of the acoustic feature analysis were the same as those of the Chinese data.

The evaluation experiments for these techniques were carried out as follows. First, we performed MLE-based HMM training on the Chinese and Japanese data using HTK tools (Young et al., 1999), and the segmentation results were regarded as the baseline result. MSGE-based discriminative training and explicit duration modeling were applied respectively and the segmentation accuracy was compared to the baseline. Finally, we examined the effect of the combination of these two techniques on HMM-based segmentation.

4.2. MSGE-based HMM training

4.2.1. Effect on Chinese corpus

From the result of the closed and open tests in Fig. 5(a), MSGE-based discriminative training is gradually convergent after 10–20 iterations. As can be seen in Table 1, the accuracy of segmentation is improved after MSGE-based discriminative
training, especially for errors greater than 5 ms. We also examined the effect of error length on the loss function by training the HMMs with different $\alpha$ values. When $\alpha$ increases from 0 to 1, which means that we have more focus on larger errors, the percentage of error greater than 30 ms decreased by 0.13, whereas the percentage of error greater than 5 ms increased by 0.83.

The details on accuracy with different phonetic boundaries are shown in Table 2. After the MSGE-based discriminative training, the average error of the CV-boundary decreased from 4.51 ms to 3.60 ms, i.e., a reduction of 19.7%, whereas that of the VV-boundary decreased 9.6%. Since we noted that concatenative speech synthesis is much more sensitive to the accuracy of the CV-boundary and insensitive to the VV-boundary, this improvement appears to be reasonable for speech synthesis.

### 4.2.2. Effect on Japanese corpus

The convergence of MSGE-based discriminative training on Japanese data can be found in Fig. 5(b). As seen in Table 3, the segmentation accuracy on Japanese was improved after MSGE-based training, and the effect of error length with different $\alpha$ values is similar to that on Chinese. In Table 4, the largest improvement also occurred on the accuracy of the CV-boundary, where the error average dropped from 7.85 ms to 4.84 ms, i.e., a 38% reduction.

Comparing the results for Japanese and Chinese data, we found that the improvement for Japanese data is more significant than for Chinese data.

### Table 1
Effect of MSGE-based training on Chinese

<table>
<thead>
<tr>
<th>Percentage of error (%)</th>
<th>Aver (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>&gt;5 ms</td>
</tr>
<tr>
<td>Baseline</td>
<td>29.56</td>
</tr>
<tr>
<td>MSGE ($\alpha = 0$)</td>
<td>23.99</td>
</tr>
<tr>
<td>MSGE ($\alpha = 1$)</td>
<td>24.82</td>
</tr>
<tr>
<td>EDM</td>
<td>26.87</td>
</tr>
<tr>
<td>EDM&amp;MSGE</td>
<td>23.56</td>
</tr>
</tbody>
</table>

### Table 2
Accuracy on different phonetic boundary (Chinese)

<table>
<thead>
<tr>
<th>Error average (ms)</th>
<th>CC</th>
<th>CV</th>
<th>VC</th>
<th>VV</th>
</tr>
</thead>
<tbody>
<tr>
<td>Baseline</td>
<td>×</td>
<td>4.51</td>
<td>5.69</td>
<td>11.99</td>
</tr>
<tr>
<td>MSGE ($\alpha = 0$)</td>
<td>×</td>
<td>3.60</td>
<td>5.37</td>
<td>10.83</td>
</tr>
<tr>
<td>EDM</td>
<td>×</td>
<td>4.31</td>
<td>5.20</td>
<td>10.57</td>
</tr>
<tr>
<td>EDM&amp;MSGE</td>
<td>×</td>
<td>3.63</td>
<td>4.99</td>
<td>9.89</td>
</tr>
</tbody>
</table>

C: consonant, V: vowel.

### Table 3
Effect of MSGE-based training on Japanese

<table>
<thead>
<tr>
<th>Percentage of error (%)</th>
<th>Aver (ms)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>&gt;5 ms</td>
</tr>
<tr>
<td>Baseline</td>
<td>39.16</td>
</tr>
<tr>
<td>MSGE ($\alpha = 0$)</td>
<td>29.85</td>
</tr>
<tr>
<td>MSGE ($\alpha = 1$)</td>
<td>30.32</td>
</tr>
<tr>
<td>EDM</td>
<td>34.54</td>
</tr>
<tr>
<td>EDM&amp;MSGE</td>
<td>30.05</td>
</tr>
</tbody>
</table>
is much larger than that for Chinese. One reason for this is that the HMM modeling in Japanese is not optimized, that is, it simply uses a 3-state model for all phonemes. Therefore, the segmentation accuracy of the baseline trained by MLE criteria for Japanese is much worse than that for Chinese. Nevertheless, the difference in accuracy between Japanese and Chinese data is reduced after MSGE-training. This indicates that the MSGE-based training method can work well even when the HMM modeling is not optimized. Furthermore, it can compensate for the inaccuracy of the HMM modeling to a certain extent.

4.3. Explicit duration modeling (EDM)

From the results, segmentation accuracy was improved after the MSGE-based discriminative training. However, the improvement in eliminating large errors, e.g., larger than 30 ms, is very limited. To reduce large errors, explicit duration modeling was applied with weight optimization. The experiments were performed both on Chinese and Japanese data, and the results are shown in Tables 1–4.

As can be seen in Table 1, segmentation accuracy was improved after applying explicit duration modeling, and the greatest improvement is that the number of errors larger than 30 ms had a 27% reduction. As large errors are usually related to the VV boundary, the error average of the VV boundary showed the greatest improvement, from 11.99 ms to 10.57 ms in Table 2. Similar results for Japanese data can be found in Tables 3 and 4, where the number of errors larger than 30 ms was reduced 47%, and the error average of the VV boundary was improved from 11.31 ms to 9.23 ms. In fact, if we regard segmentation errors larger than 30 ms as “real” errors, i.e., ones large enough to likely cause degradation in synthesis quality, the improvement of the segmentation accuracy is rather impressive.

By comparing the results on Japanese and Chinese, we found that the effect of duration modeling on Japanese is much larger than that on Chinese. A possible reason is that the phone duration in Japanese is more stable than that in Chinese, i.e., in the different context, the variation of the duration for a given Japanese phoneme is less than that for a given Chinese phoneme. This indicates that the effect of explicit duration modeling is language dependent. Nevertheless, segmentation performance was greatly improved after applying explicit duration modeling, which focuses on eliminating large errors.

4.4. Combination of two techniques

From the evaluation results of MSGE-based training and explicit duration modeling, these two techniques can improve segmentation accuracy with different focuses, where MSGE-based discriminative training focuses on improving the accuracy of the sensitive boundary, e.g., the boundary between plosive and vowel, and explicit duration modeling focuses on eliminating large errors. Finally, we combined both techniques to improve segmentation accuracy. The results on Chinese and Japanese data are shown in Tables 1–4.

In Table 1, the percentage of errors greater than 5 ms dropped to 23.56%, which is similar to the result after MSGE-based training, and the percentage of errors greater than 30 ms dropped to 1.68%, which is similar to the result after explicit
duration modeling. From Table 2, we can see that the final results for each boundary type were improved. Also, similar results for Japanese data can be found in Tables 3 and 4. By comparing the results with those of previous experiments, the final results have the advantages of both MGE-based discriminative training and explicit duration modeling, i.e., the segmentation accuracy of the sensitive boundary showed considerable improvement and the number of large errors was largely reduced. The final average errors of segmentation on Chinese and Japanese are 5.79 ms and 6.61 ms respectively, which is much better than that of the baseline.

5. Discussion

It can be seen from the experiment results that segmentation accuracy was greatly improved after applying the discriminative training and the explicit duration modeling, and the improvement is consistent with the requirement of concatenative speech synthesis. As the quality of corpus-based concatenative speech synthesis is directly related to the accuracy of the phonetic labeling of the corpus, the improvement of segmentation accuracy should lead to the improvement of the synthesis quality.

However, the requirement of segmentation accuracy varies for different speech synthesis system, and synthesis quality is also affected by other important factors, e.g., unit selection algorithm and postprocessing technique of the concatenative units, etc. In (Chu et al., 2001), the TTS system selects units based on the prosodic and phonetic features and concatenates them without any modification. Therefore, segmentation accuracy of the corpus is critically important for this system, and the improvement of segmentation accuracy will bring the improvement of synthesis quality.

In (Huang et al., 1996), Microsoft trainable TTS system, Whisler, automatically prunes the outliers from the corpus and selects a number of optimal candidates for each unit based on HMM scores. In (Hunt and Black, 1996), ATR’s TTS system CHATR attempts to correct possible segmentation error by searching for the best concatenative point in the vicinity of the unit boundary at run time. In these two TTS system, the degradation of synthesis quality caused by segmentation errors can be alleviated through two different ways, removing or revising the units which are segmented by error. Therefore, it is difficult to estimate how much the improvement of synthesis quality would be after using these two techniques to improve segmentation accuracy.

In another view, the time used for searching the best concatenative point in CHATR system could be reduced if segmentation accuracy is improved, hence results in improved synthesis efficiency. Also for Whisler system, the outliers could be smaller, and the HMM scores should be more reliable. Therefore, it is believable the improvement of segmentation accuracy will be beneficial for concatenative speech synthesis system in different aspects, not limited in synthesis quality.

6. Summary

In this paper, the discriminative training method is applied to train HMMs for the segmentation task, and a minimum segmentation error (MGE) based criterion is proposed. In this method, a loss function directly related to the segmentation error is defined, and parameter optimization is performed by the Generalized Probabilistic Descent (GPD) algorithm. To eliminate large errors, we applied explicit duration models and proposed a two-step-based segmentation method to solve the problem of computational cost, where the duration model is incorporated in a postprocessor procedure. From the experimental results, these two techniques significantly improved the segmentation accuracy with different focuses, where MGE-based training focuses on improving the accuracy of the sensitive boundary, e.g., the boundary between the plosive and vowel, and explicit duration modeling focuses on eliminating large errors. After combining these two techniques, the segmentation accuracy was improved with both advantages, and the final average errors of segmentation on Chinese and Japanese were 5.79 ms and 6.61 ms respectively, which is much better than the baseline.
Acknowledgement

The research reported here was supported in part by a contract with the Telecommunications Advancement Organization of Japan entitled “A study of speech dialogue translation technology based on a large corpus”.

References