Front. Electr. Electron. Eng. China 2010, 5(1): 65–71
DOI 10.1007/s11460-009-0076-0

RESEARCH ARTICLE

**Hao WU, Xihong WU, Huisheng CHI**

# Product HMM-based training method for acoustic model with multiple-size units



**Abstract** Multiple-size units-based acoustic modeling has been proposed for large vocabulary speech recognition system to improve the recognition accuracy with limited training data. By introducing a limited number of long-size units into unit set, this modeling scheme can make better acoustic model precision than complete short-size unit modeling without losing model trainability. However, such a multiple-size unit acoustic modeling paradigm does not always bring reliable improvement on recognition performance, since when a large number of long-size units are added in, the amount of training data for short-size units will decrease and result in insufficiently trained models. In this paper, a modified Baum-Welch training method is proposed, which uses product hidden Markov models (PHMMs) to couple units with different sizes and enables them to share same portions of training data. The validity of proposed method is proved by experiment results.

**Keywords** multiple-size unit, product hidden Markov model (PHMM), product of exports

## 1 Introduction

Unit selection is one of the most important issues in acoustic modeling for automatic speech recognition (ASR). In the past decades, different types of basic acoustic model units have been evaluated. Generally speaking, with appropriate model structures, long-size units can capture larger amount of phonological phenomena (e.g., long-span coarticulation) and implicitly resolve or align events in the speech with events in the sequence of concatenated short units, and thus have better precision over short-size units [1–3]. However, compared with short-size units, long-size units modeling scheme suffers greatly from training data sparsity and lexicon generalization problem; thus, it is hardly practiced in large vocabulary continuous speech recognition systems [1,2].

In dealing with these problems, multiple-size units-based modeling schemes have been proposed in several works. Usually, a hybrid units set is constructed by appending a limited number of longer-size units into a self-contained short-size unit set. The type of selected long-size units can be specified according to the characteristics of language and the condition of the task. In Ref. [2], such multiple-size unit sets were constructed by appending syllables and monosyllabic words into the short-size unit set, while in Ref. [3], such set is constructed by appending syllables and vowel-consonant units. In our recent studies [4], context-dependent syllables are selected as the complementary long-size units. In most of these implementations, the occurring frequency of the units in the training data is used as the criteria for selection.

Baum-Welch algorithm [5,6] has been widely used for training acoustic models. In the general Baum-Welch training scenario, one composite hidden Markov model (HMM) is assembled for each utterance and then used in forward-backward propagation for statistics accumulating. Usually, the composite HMM is constructed by joining the HMMs of the phonetic units in sequence corresponding to the symbolic transcription of the training utterance.

When training multiple-size unit models, each symbol in the transcription can be phonetically represented by either long-size units or short-size units. Since only one phonetic representation is used for each symbol to construct the composite HMM, the corresponding portions of the data will only be used for training the models for selected representing units and no longer available for those units in other representations.

In traditional training approach for multiple-size unit acoustic modeling, the transcription is usually represented by long units wherever they could be used; so the training data for long-unit model are sufficient. As a result, the training data for short-size unit model are decreased. This

Received January 14, 2009; accepted October 3, 2009

Hao WU, Xihong WU (✉), Huisheng CHI
Hearing Research Center, Key Laboratory of Machine Perception (Ministry of Education), Peking University Beijing 100871, China
E-mail: wxh@cis.pku.edu.cn

may cause few problems when the number of appended long-size units is small or the training data are sufficient. However, when a lot of high-frequency long-size units are appended into the unit set, such “training data contest” may bring a lot of less sufficiently trained short-size unit models and may induce significant degeneration on the recognizing performance [4].

To solve this problem, the general strategy is to share the same portion of data in training both long-size unit model and corresponding short-size unit models. Moreover, for keeping consistency, two kinds of models should be coupled together with a composite model structure so that their parameters can be jointly estimated.

Word lattice is a compact structure for embedding multiple representations and has been used in statistic calculation for discriminate training [7]. However, as will be discussed later, applying Baum-Welch on a lattice embedding multiple-size units may cause the problem of “miscalculation of statistics”. In this paper, we use product hidden Markov model (PHMM) instead of lattice to couple multiple phonetic representations. A corresponding variation of Baum-Welch algorithm is also proposed for parameter estimation.

In Sect. 2, the principle of the method is presented. Experimental setup and performance evaluation are introduced in Sect. 3. Section 4 shows the experimental results, followed by discussion on the results. Section 5 gives the conclusions and outlines future works.

## 2 Training method

Word lattice can be used to embed different phonetic representations for each transcription symbol, as illustrated in Fig. 1, where the long-size unit model “d_e0” (pronunciation for Chinese character “的”) and the corresponding short-size units’ model joining with initial “d” and final “e0” are embedded. The structure of such lattice seems similar to the one which is used in training acoustic models with pronunciation variants.

However, when using Baum-Welch algorithm, the word lattice being used in pronunciation variants training may cause underlying problem in multiple-size unit model training, since there are essential differences between these two training problems.

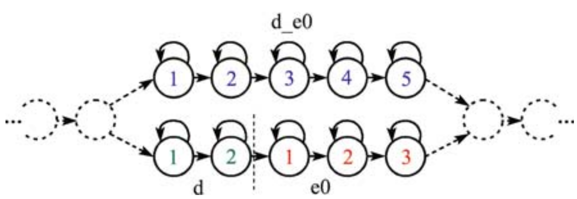


**Fig. 1** Lattice containing two phonetic representations with different unit size

For pronunciation variants training problem, the different phonetic variants are essentially exclusive to each other, that is, only one variant is supposed to be implemented as the surface form. On the other hand, when training models with multiple-size units, different phonetic representations are actually regarded as describing the same training data with different types of models (different phonetic scales or time spans, etc.). That is, all the representations are implemented consubstantially in the same portion of data.

Such difference makes it necessary to bring some extra consideration when using Baum-Welch algorithm in multiple-size units training, which is mainly related to the definition and calculation of posterior probability $\gamma_t(j)$.

In a general Baum-Welch training algorithm, the term of posterior probability $\gamma_t(j)$ defined by the following equation gives the probability of occupying state $j$ at time $t$ with observation sequence $X$:

$$\gamma_t(j) = \frac{p(X, s_t = j|\Theta)}{p(X|\Theta)}, \tag{1}$$

where $s_t$ is the identity of the state occupying at time $t$, and $\Theta$ is the current parameter set. Given this term, new parameters (means, covariance matrices, and component weights for each state) are simply computed by summing up the relevant statistics weighted by $\gamma_t(j)$ over all training utterances [1].

With the difference mentioned above, the constraint on the distribution of $\gamma_t(j)$ is changed as shown in Fig. 2.

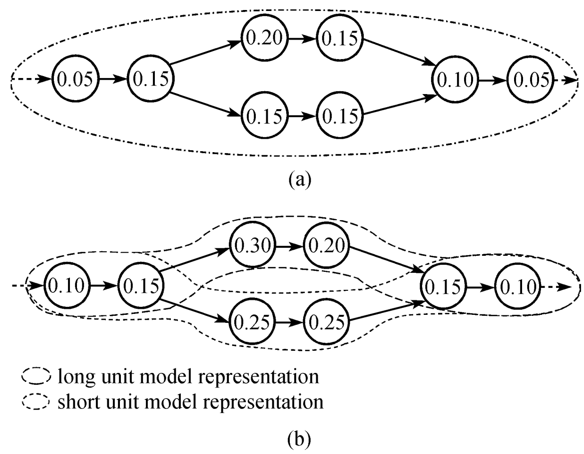


**Fig. 2** Distribution of posterior probability $\gamma_t(j)$ in different training problems. (a) Original training scenario; (b) multiple-size unit model training

In original training scenario (Fig. 2(a)), at each time, for each utterance, $\gamma_t(j)$ is distributed over all the states, so the sum-to-one rule of posterior probability is applied to all the states in lattice $(0.05 + 0.15 + 0.20 + 0.15 + 0.15 + 0.15 + 0.10 + 0.05 = 1.00)$. While training multiple-size unit model (Fig. 2(b)), $\gamma_t(j)$ should be distributed over the states within each phonetic representation path, so the sum-to-one rule is applied to each of two different state sets (states from long-unit model and states from short-unit model) in the lattice $(0.10 + 0.15 + 0.30 + 0.20 + 0.15 + 0.10 = 0.10 + 0.15 + 0.25 + 0.25 + 0.15 + 0.10 = 1.00)$.

The lattice-based structure in Fig. 1 can be used in Baum-Welch algorithm and result in good convergence. However, during the forward-backward process, the constraint shown in Fig. 2(b) is not kept, and $\gamma_t(j)$ for each representation is underestimated. For short-size unit model, this brings the model an inappropriate favor towards those data which are not shared by longer-size units.

Different from word lattice structure, PHMM focuses on representing observation sequence the consubstantial processes. PHMM has been studied in the area of audiovisual speech recognition [8–10]. In this model, each state is built by merging an $N$-tuple ($N = 2$ here) of states from the different phonetic representations of the composite model for transcription symbols with multiple-size phonetic representations. The topologies of PHMM are defined so as to represent all the possible state paths given the initial HMM topologies for each phonetic representation. This model allows implementing independent search within subunits as well as intraunits synchrony constraints. Figure 3 shows PHMM model obtained from the lattice structure in Fig. 1.

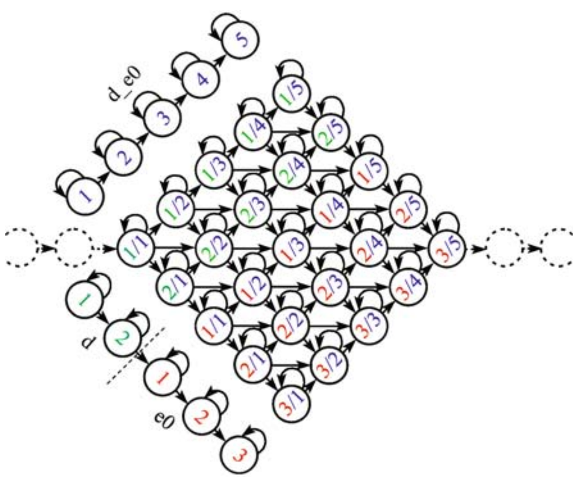


**Fig. 3** PHMM network for multiple-size model training

PHMM is originally proposed to model synchronous streams generated by different sources. By properly assigning the output distribution for each composite state, PHMM can be used in Baum-Welch algorithm and generate more accurate statistics for parameter updating, while keeping the constraint in Fig. 2(b). Most PHMM implementations use the product of stream likelihoods as the output distribution for composite states. Under the assumption of observation streams independency, the likelihood for an observation vector $\boldsymbol{x}_t$ on composite state $i$ is composed as the weighted product of component states $i_S$ from each individual stream $S$:

$$p(\boldsymbol{x}_t|i) = \prod_S [p(\boldsymbol{x}_{S,t}|i_S)]^{\lambda_S}. \tag{2}$$

However, when PHMM is used in multiple-size units model training, different models will be used to represent the same features; thus, the product of probabilistic density functions (PDFs) is no longer a valid PDF. One general solution to such problem is the well-known products of experts (PoEs) [11,12] framework, in which PDF $p(\boldsymbol{x}_t|\Theta)$ is computed as a normalized multiplication of probabilities $p(\boldsymbol{x}_t|\Theta)$ of $N$ individual experts with parameters $\Theta_1, \Theta_2, \ldots, \Theta_N$:

$$p(\boldsymbol{x}_t|\Theta) = \frac{\prod_{n=1}^{N} p(\boldsymbol{x}_t|\Theta_n)}{\int_{\boldsymbol{x}'} \prod_{n=1}^{N} p(\boldsymbol{x}'|\Theta_n)\mathrm{d}\boldsymbol{x}'}. \tag{3}$$

Most HMM-based speech recognition systems use mixture of Gaussians to model the state distribution. A PoE of such distributions (referred to as a product of mixture of Gaussians (PoMoG)) can be analytically normalized and thus be used as an alternative state distribution [12]. By this way, when a composite state $i$ is composed from long-size unit model state and short-size unit model state, the distribution for $i$ can be expressed as

$$p(\boldsymbol{x}_t|i) = \frac{1}{Z} \prod_{u \in \{\text{long},\text{short}\}} \left[ \sum_{m=1}^{M} c_{u,m} N(\boldsymbol{x}_t; \boldsymbol{\mu}_{u,m}, \boldsymbol{\Sigma}_{u,m}) \right], \tag{4}$$

where the normalization term $Z$ is computed with the parameters from the corresponding component states. This distribution can be expressed as a mixture model in a product space [12]:

$$p(\boldsymbol{x}_t|i) = \frac{1}{Z} \sum_{\boldsymbol{m}} c_{\boldsymbol{m}} K_{\boldsymbol{m}} N(\boldsymbol{x}_t; \boldsymbol{\mu}_{\boldsymbol{m}}, \boldsymbol{\Sigma}_{\boldsymbol{m}}), \tag{5}$$

where $\boldsymbol{m} = [m_{\text{long}}, m_{\text{short}}]'$ determines a metacomponent in which $m_{\text{long}}$ and $m_{\text{short}}$ specify the component from states in either the long-unit model or the short-unit model. $K_{\boldsymbol{m}}$ is an observation-independent normalization which can be expressed as

$$K_{\boldsymbol{m}} = \frac{(2\pi)^{\frac{d}{2}} |\boldsymbol{\Sigma}_{\boldsymbol{m}}|^{\frac{1}{2}}}{\prod_{u \in \{\text{long},\text{short}\}} (2\pi)^{\frac{d}{2}} |\boldsymbol{\Sigma}_{m_u}|^{\frac{1}{2}}} \times \mathrm{e}^{\frac{1}{2}\boldsymbol{\mu}'_{\boldsymbol{m}} \boldsymbol{\Sigma}_{\boldsymbol{m}}^{-1} \boldsymbol{\mu}_{\boldsymbol{m}} - \sum_{u \in \{\text{long},\text{short}\}} (\boldsymbol{\mu}'_{m_u} \boldsymbol{\Sigma}_{m_u}^{-1} \boldsymbol{\mu}_{m_u})}. \tag{6}$$

The summation in this form is over all metacomponent combinations. The mean, covariance matrix, and prior (component weight) may be expressed as

$$\boldsymbol{\mu}_{\boldsymbol{m}} = \boldsymbol{\Sigma}_{\boldsymbol{m}} \left( \sum_{u \in \{\text{long, short}\}} \boldsymbol{\Sigma}_{m_u}^{-1} \boldsymbol{\mu}_{m_u} \right), \tag{7}$$

$$\boldsymbol{\Sigma}_{\boldsymbol{m}} = \left( \sum_{u \in \{\text{long, short}\}} \boldsymbol{\Sigma}_{m_u}^{-1} \right)^{-1}, \tag{8}$$

$$c_{\boldsymbol{m}} = \prod_{u \in \{\text{long,short}\}} c_{m_u}. \tag{9}$$

The effective number of metacomponents $M$ is the number of combinations of components from each of corresponding long-unit model state and short-unit model state. The normalization term $Z$ in Eq. (5) can be written as

$$Z = \sum_{\boldsymbol{m}} c_{\boldsymbol{m}} K_{\boldsymbol{m}}. \tag{10}$$

The detailed derivation can be found in Ref. [12].

Typically, with two HMMs containing $n_1$ and $n_2$ states, the number of composite states in the combined PHMM is $n_1 \times n_2$. By applying an asynchrony limit on the model topology, the number of product model states can be efficiently decreased, as illustrated in Fig. 4. In our implementation, the asynchrony width is set to 1 (only 1 state of asynchrony is allowed).

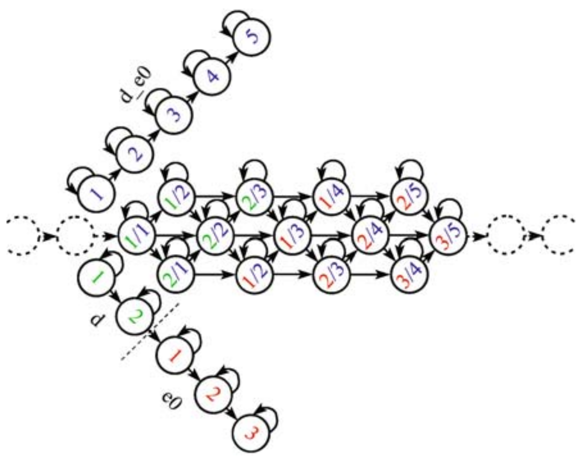


**Fig. 4** PHMM network with path constraint applied

Also, by compiling PHMM with more parallel HMMs, these methods can be extended to combine the units with more different sizes (e.g., word).

In our experiments, the parameters of PHMM are obtained by expectation maximization (EM) algorithms [8]. Given the definition of observation likelihood $p(\boldsymbol{x}_t|i)$ in Eq. (4), $\gamma_t$ can be directly computed through the standard forward-backward propagation. Given this $\gamma_t$ computation, the whole algorithms are the same as those in Baum-Welch algorithm, except that in the parameter updating equation, the corresponding statistics are accumulated on each component of the composite state. It should be pointed out that although the parameter estimation method mentioned above has made stable model convergence and gives fine performance in our experiments, the strict theoretical validity of this method is still an ongoing work, since there are theoretical difficulties related to the normalization term in PoE.

## 3 Experimental setup

In this section, a series of speech recognition tasks are carried out on Chinese continuous speech evaluation data to evaluate the performance of the method. In our implementation, Chinese syllables are selected as long-size units, while Chinese initials and finals are selected as short-size units.

### 3.1 Training data

In our experiments, all the models are trained with a speech database which contains about 360 hours speech of over 750 male speakers. The speech data in this corpus are picked up from three widely used continuous Mandarin speech corpora: 863-I, 863-II, and Intel corpora. The brief information about these speech corpora is listed in Table 1.

**Table 1** Training corpora

| corpus | speakers | amount of speech/h |
|---|---|---|
| 863-I (male) | 83 | 56.67 |
| 863-II (male) | 120 | 78.08 |
| Intel (male) | 556 | 227.30 |
| total | 759 | 362.05 |

### 3.2 Training procedures

The acoustic features used here are the Mel-frequency cepstral coefficients (MFCCs), which include 12 cepstral coefficients together with the logarithmic energy. Histogram equalization-based cepstral normalization is applied to compensate the convolving and additive noise. By including the first and second derivatives, the feature vectors have 39 elements. In our experiments, three unit sets are constructed, including a baseline initial-tonal final set (ITF-BL) and two multiple-size units sets.

The purpose of introducing a baseline initial-final set is to inspect the goodness of the initial and final models in multiple-size unit sets with different training methods, since the motivation of this research is to guarantee the training of short-size unit models while improving the training of long-size unit models.

By adding four dummy initials (“ga”, “go”, “ge”, “er”), two semivowels (“y”, “w”), and some neutral tonal finals to the standard ITF set, ITF-BL set includes 27 initials and 187 tonal finals. When constructing a multiple-size unit set, all the units in ITF-BL are included firstly, and then several additional high-frequency syllables are appended to the set.

In our experiments, two multiple-size units sets were constructed according to the statistics from training corpus. The first set (H40) includes all initial-final units and 40 most frequently occurred syllables. These 40 syllables hold about 25% occurrences of syllables in the whole corpus. The second set (H100) includes all the initial-final units and 100 most frequently occurred syllables. These 100 syllables hold over 50% occurrences of syllables in the whole corpus. The added high-frequency syllables are listed in Table 2.

**Table 2** Top 40 and 100 high-frequency syllables (tonal syllables) (1 k = 1000 times)

| rank | syllable | occurs | rank | syllable | occurs |
|---|---|---|---|---|---|
| top 40 high frequency syllables | | | 31 | zhi4 | 33 k |
| (included in H40 and H100) | | | 32 | ji4 | 33 k |
| 1 | de0 | 231 k | 33 | chu1 | 32 k |
| 2 | shi4 | 125 k | 34 | zhe4 | 31 k |
| 3 | yi4 | 80 k | 35 | di4 | 31 k |
| 4 | shi2 | 70 k | 36 | lai2 | 30 k |
| 5 | zai4 | 68 k | 37 | zhi1 | 30 k |
| 6 | bu4 | 66 k | 38 | er4 | 30 k |
| 7 | you3 | 59 k | 39 | dui4 | 30 k |
| 8 | ren2 | 59 k | 40 | yi2 | 30 k |
| 9 | yi1 | 51 k | the 41th–100th high frequency | | |
| 10 | le0 | 50 k | syllables (included in H100) | | |
| 11 | yuan2 | 47 k | 41 | yao4 | 29 k |
| 12 | he2 | 47 k | 42 | ji2 | 29 k |
| 13 | zhong1 | 46 k | 43 | li3 | 29 k |
| 14 | guo2 | 46 k | 44 | zuo4 | 29 k |
| 15 | er2 | 44 k | 45 | sheng1 | 29 k |
| 16 | ta1 | 43 k | 46 | xian4 | 29 k |
| 17 | yi3 | 42 k | 47 | nian2 | 29 k |
| 18 | da4 | 41 k | 48 | xin1 | 28 k |
| 19 | cheng2 | 39 k | 49 | yu3 | 27 k |
| 20 | li4 | 39 k | 50 | jing1 | 27 k |
| 21 | gong1 | 38 k | 51 | jian4 | 27 k |
| 22 | jia1 | 38 k | 52 | ji1 | 26 k |
| 23 | yu2 | 38 k | 53 | jin4 | 26 k |
| 24 | hui4 | 36 k | 54 | wu3 | 26 k |
| 25 | wei4 | 36 k | 55 | zheng4 | 26 k |
| 26 | dao4 | 35 k | 56 | xing2 | 26 k |
| 27 | wei2 | 35 k | 57 | hou4 | 26 k |
| 28 | wo3 | 34 k | 58 | ye4 | 25 k |
| 29 | shang4 | 34 k | 59 | yue4 | 24 k |
| 30 | ge4 | 34 k | 60 | ming2 | 24 k |

(*Continued*)

| rank | syllable | occurs | rank | syllable | occurs |
|---|---|---|---|---|---|
| 61 | ye3 | 24 k | 81 | men0 | 20 k |
| 62 | hua4 | 24 k | 82 | fang1 | 20 k |
| 63 | you2 | 23 k | 83 | mei3 | 20 k |
| 64 | hao4 | 23 k | 84 | guan1 | 19 k |
| 65 | xiang4 | 23 k | 85 | ke3 | 19 k |
| 66 | xia4 | 23 k | 86 | zhi3 | 19 k |
| 67 | fa1 | 23 k | 87 | yan2 | 19 k |
| 68 | yong4 | 23 k | 88 | qi2 | 19 k |
| 69 | jiu4 | 22 k | 89 | fu4 | 19 k |
| 70 | ri4 | 22 k | 90 | xi1 | 18 k |
| 71 | si1 | 22 k | 91 | zi4 | 18 k |
| 72 | duo1 | 22 k | 92 | min2 | 18 k |
| 73 | wai4 | 22 k | 93 | she4 | 18 k |
| 74 | yin1 | 22 k | 94 | fen1 | 18 k |
| 75 | neng2 | 22 k | 95 | bei4 | 18 k |
| 76 | qian2 | 22 k | 96 | jin1 | 18 k |
| 77 | san1 | 21 k | 97 | chang3 | 18 k |
| 78 | quan2 | 21 k | 98 | bu2 | 18 k |
| 79 | jiang1 | 21 k | 99 | you4 | 18 k |
| 80 | yu4 | 21 k | 100 | wu4 | 18 k |

Figure 5 shows some most significant drop-off of the training tokens (“e0”, “ib4”, “i4”, “uo2”, “ong1”, “van2”) for tonal final units in H40 unit set when syllable unit is used to represent all the corresponding training data.

Left-to-right HMM topology is used to represent each unit, and one state skipping is allowed within each model. The number of states in each model was set to 2–3 for initials and 4–5 for tonal finals, according to the corresponding phonetic structures. For high-frequency syllables, the number of states is set as the sum of the number of states in the corresponding initial and final models.

We trained the models with the proposed method. In all training schemes, acoustic model units are represented by continuous mixture density HMMs, the proposed method is used to estimate the parameters of the models, and decision-tree-based state tying is used to reduce the total number of parameters. Number of tied states is set to 8000, 10000, and 12000 for each training scheme.

### 3.3 Pronunciation generation

When using acoustic models with multiple-size units in large vocabulary speech recognition, a pronunciation dictionary containing different size units has to be generated. Replacing all shorter-size units with longer-size units in the dictionary does not always bring benefit to the performance. In our implementation, the dictionary is generated with a baseform-selecting method proposed by Ref. [3].

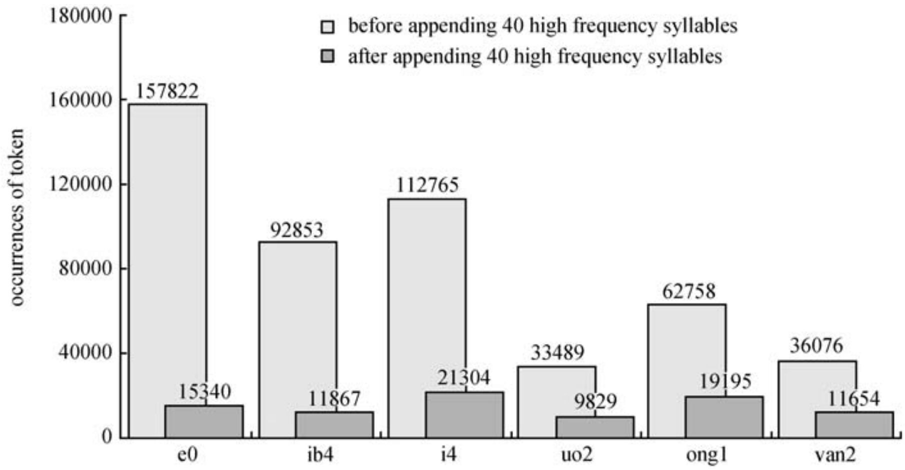


**Fig. 5** Most significant decrease of training tokens in training data

## 4 Experimental results and discussion

In this paper, we took large vocabulary reorganization as the evaluating task. The evaluation data used in these experiments are clean (defined as the "F0" condition[1)]) male speech data from the evaluation set of HUB-4 NE Chinese broadcast news corpus. This evaluation set includes 238 sentences uttered by male speakers, which sum up to 15 minutes. Before the experiment on evaluation set, parameters of the decoder are optimized with the clean male speech from the HUB-4 NE development set. The results are shown in Table 3.

It can be observed from Table 3 that with enough tied states, multiple-size unit models achieve better performance over the baseline ITF models, which proves the validity of multiple-size units modeling. Further comparison shows that the proposed training method gets better performance than using straightforward Baum-Welch training framework. With the number of appended syllable increasing, this predominance becomes more significant.

To further investigate the influence on short-size unit models by different training methods, another experiment is carried out in which only the initial-final models in each model (even when using the models with multiple-size units) are used in recognition. ITF-BL model with 8000 tied states is used as the baseline, while H40 models with 10000 and H100 models with 12000 tied states are selected for comparison, since the number of tied states for shorter-size units in these models is close to 8000. The results are shown in Fig. 6.

It could be observed that the training of multiple-size unit models did bring degeneration over initial and final models. This degeneration is remarkable in the case of straightforward Baum-Welch training and mitigated in the proposed method, which further proves that with the proposed method, the mentioned data-depriving problem is alleviated. The remaining gaps of the performance between the ITF-BL model and PHMM models indicate that PoMoG may not be accurate enough for representing the likelihood of the composite state.

## 5 Conclusions and future work

In this paper, we propose a PHMM-based training method for handling the problem of depriving training data for short-size unit model in multiple-size acoustic model

**Table 3** Chinese character error rate (substitution + deletion + insertion) with different training methods

| unit set | training methods | word error rate/% | | |
|---|---|---|---|---|
| | | number of tied states = 8000 | number of tied states = 10000 | number of tied states = 12000 |
| ITF-BL | Baum-Welch | 17.60 | 18.03 | 18.11 |
| H40 | Baum-Welch | 18.02 | 17.64 | 17.68 |
| | PHMM Baum-Welch | 17.82 | 17.48 | 17.33 |
| H100 | Baum-Welch | 20.03 | 19.86 | 19.02 |
| | PHMM Baum-Welch | 18.75 | 17.62 | 17.03 |

1) The 1996 "Hub-4" annotation specification for evaluation of speech recognition on broadcast news. ftp://jaguar.ncsl.nist.gov/csr96/h4/h4annot.ps

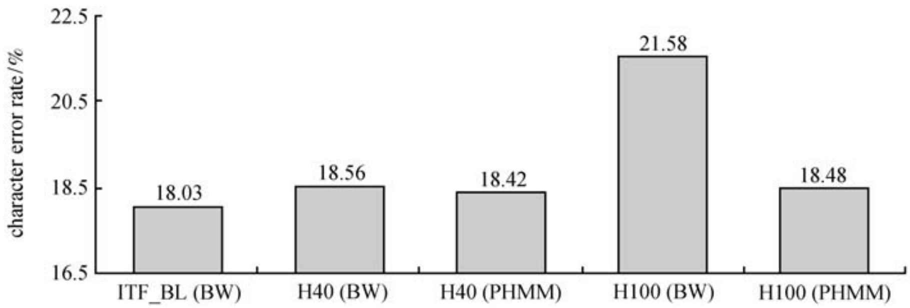


**Fig. 6** Character error rates for using only short-size unit models

training. Experiments show that the proposed method can make improvement on the performance of large vocabulary recognition for continuous Chinese speech.

Next, we are going to focus on more theoretical analysis for the convergence problem. The corresponding discriminative training methods (e.g., maximum mutual information estimation (MMIE) or minimum phone error (MPE)) and alternative composite state distribution representations will also be investigated.

**Acknowledgements** This work was supported in part by the National Natural Science Foundation of China (Grant No. 60605016), the National Key Basic Research Program of China (Nos. 2004CB318005 and 2004CB318105), and the National High Technology Research and Development Program of China (No. 2006AA010103).

## References

1. Huang X D, Acero A, Hon H W. Spoken Language Processing. New Jersey: Prentice Hall, 2000
2. Ganapathiraju A, Hamaker J, Picone J, Ordowski M, Doddington G R. Syllable-based large vocabulary continuous speech recognition. IEEE Transactions on Speech and Audio Processing, 2001, 9(4): 358–366
3. Nagarajan T, Vijayalakshmi P, O'Shaughnessy D. Combining multiple-sized sub-word units in a speech recognition system using baseform selection. In: Proceedings of Interspeech 2006—Ninth International Conference on Spoken Language Processing. 2006, 1595–1597
4. Wu H, Wu X H. Context dependent syllable acoustic model for continuous Chinese speech recognition. In: Proceedings of Interspeech 2007. 2007, 1713–1716
5. Baum L E, Petrie T, Soules G, Weiss N. A maximization technique occurring in the statistical analysis of probabilistic functions of Markov chains. Annals of Mathematical Statistics, 1970, 41(1): 164–171
6. Juang B H, Levinson S, Sondhi M. Maximum likelihood estimation for multivariate mixture observations of Markov chains. IEEE Transactions on Information Theory, 1986, 32(2): 307–309
7. Povey D. Discriminative training for large vocabulary speech recognition. Dissertation for the Doctoral Degree. Cambridge: Cambridge University, 2004
8. Nefian A, Liang L, Pi X, Liu X, Murphy K. Dynamic Bayesian networks for audio-visual speech recognition. EURASIP Journal on Applied Signal Processing, 2002, 2002(11): 1274–1288
9. Tomlinson M J, Russell M J, Brooke N M. Integrating audio and visual information to provide highly robust speech recognition. In: Proceedings of 1996 IEEE International Conference on Acoustics, Speech, and Signal Processing. 1996, 821–824
10. Gravier G, Potamianos G, Neti C. Asynchrony modeling for audio-visual speech recognition. In: Proceedings of the Second International Conference on Human Language Technology Research. 2002, 1–6
11. Hinton G E. Products of experts. In: Proceedings of the Ninth International Conference on Artificial Neural Networks. 1999, 1: 1–6
12. Gales M J F, Airey S S. Product of Gaussians for speech recognition. Computer Speech and Language, 2006, 20(1): 22–40