/

Article / Book Information

Spontaneous Speech Recognition Using a Massively Parallel Decoder

Takahiro Shinozaki *, *Sadaoki Furui*

Department of Computer Science, Tokyo Institute of Technology, Japan
{staka, furui}@furui.cs.titech.ac.jp

Abstract

Since spontaneous utterances include many variations, speaker- and task-independent general models do not work well. This paper proposes combining cluster-based language and acoustic models based on the framework of Massively Parallel Decoder (MPD). The MPD is a parallel decoder that has a large number of decoding units, in which each unit is assigned to each combination of element models. It runs efficiently on a parallel computer, and thus the turnaround time is comparable to conventional decoders using a single model and a processor. In the experiments conducted using lecture speeches from the Corpus of Spontaneous Japanese, two types of cluster models have been investigated: lecture-based cluster models and utterance-based cluster models. It has been confirmed that utterance-based cluster models give significantly lower recognition error rate than lecture-based cluster models in both language and acoustic modeling. It has also been shown that roughly 100 decoding units are enough in terms of recognition rate, and in the best setting, 12% reduction in word error rate was obtained in comparison with the conventional decoder.

1. Introduction

Recently, several large-scale spontaneous speech corpora have become available and recognition performance for spontaneous speech has been greatly improved by making speech models using a large amount of spontaneous speech data. However, recognition rates are still insufficient for the most applications. This is because spontaneous speech has many variations not only between speakers but also from utterance to utterance within each speaker. The speaker- and utterance-specific characteristics cannot be modeled by a general speaker-independent model, since they are smoothed out in the training process. One solution is to cover speech sounds by a set of speech models suitable for various input utterances.

Sentence mixture language models have been investigated in [1], in which sentence likelihood is calculated as a weighted sum of likelihood values given by component models,

$$P(W) = \sum_i \lambda_i P_i(W), \qquad (1)$$

where W is a word sequence, i is a component model index, and λ_i is a mixture weight of the i-th model. A cluster-based language model can be considered as an approximation of the mixture model replacing the summation by maximization and omitting the weight term as shown in equation (2).

$$P(W) = \max_i P_i(W). \qquad (2)$$

Similarly, a cluster-based acoustic model chooses a component model which maximizes the likelihood,

$$P(O|W) = \max_i P_i(O|W), \qquad (3)$$

where O is a sequence of observation vectors. In [2], speaker cluster-based HMMs were used to cope with the problem of frequent speaker changes in broadcast news speech recognition. Using gender dependent models in parallel can be regarded as a special case of the cluster-based modeling [3].

So far, the cluster-based models have been investigated separately for language and acoustic modeling, and they have never been combined partly because of computational cost. This paper investigates effects of the combination using a Massively Parallel Decoder (MPD). Lecture cluster-based models and utterance cluster-based models are made and evaluated using the Corpus of Spontaneous Japanese (CSJ) [4].

This paper is organized as follows. Architecture and processing time of the MPD are described in Section 2. Experimental conditions are described in Section 3 and results are presented in Section 4. Finally, some discussion are given in Section 5 and conclusions are presented in Section 6.

2. Massively Parallel Decoder

The problem of searching for the best word sequence W based on a cluster-based language model and a cluster-based acoustic model can be formulated as shown in equation (4), where i and j are element indices of the clusters, respectively. Since the order of the max operators is commutative, it can be rewritten as shown in equation (5).

$$\max_w P(O|W) P(W)$$
$$= \max_w \max_{i,j} P_i(O|W) P_j(W) \qquad (4)$$
$$= \max_{i,j} \max_w P_i(O|W) P_j(W). \qquad (5)$$

Equation (5) can be performed by first searching for $W_{i,j}$ with respect to every model combination of i and j, which can be done using a conventional decoder, and then selecting W that gives the highest likelihood.

The MPD consists of a large number of decoding units (DUs) and an integrator. It has a structure as shown in Figure 1 which corresponds to equation (5). Each DU is a conventional decoder that uses one of the combinations of element language and acoustic models. An input speech utterance is sent to all the DUs and each DU independently processes the speech based on its language- and acoustic-model. The recognition hypotheses of the DUs are gathered to the integrator and a final output is produced.

The MPD can be efficiently implemented on parallel computers such as Grid [5], MPP and SCM [6]. The Grid or the MPP connects many computers or processors to form a parallel computer, and the SCM integrates many processing units into a single chip. Parallel computers will become popular in the near feature, since they solve critical problems of single processor systems such as line delay. To take advantage of parallel computers, parallel algorithms are crucial. In this aspect, the

---

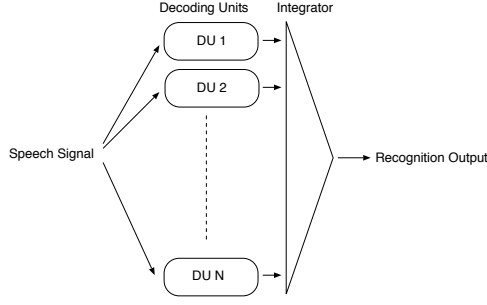*University of Washington, EE Dept., staka@u.washington.edu

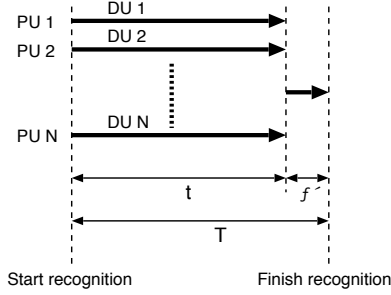Figure 1: Architecture of the Massively Parallel Decoder.

Figure 2: Processing time of MPD

MPD is well suited to parallel computers, since it has a highly parallel structure. Interactions between processing units occur only when selecting a recognition hypothesis having the highest likelihood. Therefore, the MPD can keep its high computational efficiency largely independent of the number of decoding units.

Figure 2 shows the processing time of a MPD running on a parallel computer. By assigning each DU to a different processing unit (PU), the turnaround time T of the MPD becomes constant irrespective of the number of DUs as shown in equation (6). In the equation, t and β represent processing times of the DU and the integrator, respectively. Since the processing time of the integrator is negligible compared to that of the decoding unit, equation (6) can be approximated as equation (7). Thus, the turnaround time of the MPD is about the same as conventional decoders using a single acoustic model and a single language model.

$$T = t + \beta \qquad (6)$$
$$\approx t \qquad (7)$$

3. Experimental conditions

3.1. Recognition task

The recognition task was the Test-set 1 of the Corpus of Spontaneous Japanese (CSJ). The test-set consisted of ten academic lectures given by different male speakers. In the experiment, utterances were extracted based on silence periods longer than 500ms, and five minutes of utterances were excerpted from each lecture. The subset therefore consists of 50 minutes of utterances, which corresponds to approximately a half of the Test-set 1. Figure 1 shows the lecture IDs and the number of utterances included in each five minutes set.

3.2. Acoustic models

The training set for acoustic models was the CSJ academic lectures given by male speakers, consisting of 787 lectures with the total length of 186 hours. Feature vectors had 38 elements comprising of 12 MFCC, their delta, delta delta, delta log energy

Table 1: Test-set

Conference name	# of utterances used
A01M0097	58
A04M0051	77
A04M0121	73
A03M0156	88
A03M0112	43
A01M0110	65
A05M0011	31
A03M0106	27
A01M0137	45
A04M0123	23

and delta delta log energy. The CMS (cepstral mean subtraction) was applied to each utterance. HTK [7] was utilized for model training and adaptation.

For a baseline system, a speaker-independent triphone HMM, having 3k states and 16 Gaussian mixtures in each state, was made. A regression class tree with 64 leaves was associated with the HMM for classifying Gaussian mixtures in MLLR adaptation. This model is hereafter denoted as **GAM** (General Acoustic Model).

Two types of cluster-based acoustic models were made; one was based on lecture clustering and the other was based on utterance clustering. In the utterance clustering, all the utterances were independently clustered, irrespective of the lecture in which each utterance was included. The clustering was conducted as follows.

1. Randomly assign lectures/utterances to N clusters so that all the clusters have approximately the same number of lectures/utterances. Then make each cluster-based element model.

2. Calculate likelihood of all the lectures/utterances for all the element models.

3. Re-assign lectures/utterances to clusters based on their likelihood. The assignment is constrained so that all the clusters have the same number of lectures/utterances.

4. Make a cluster-based model.

5. Return to step 2 or terminate after sufficient number of iterations.

The likelihood values were calculated using triphone label files. The number of iterations was set to 10 in the following experiments.

Based on the obtained definitions of the clusters, lecture/utterance cluster-based models for speech recognition were made by adapting the general model to each cluster using the MLLR adaptation method. These models are denoted as **LCAM(N)** (Lecture-based Cluster Acoustic Model) and **UCAM(N)** (Utterance-based Cluster Acoustic Model), where N is the number of element models.

3.3. Language models

The training set used for language modeling included academic and extemporaneous lectures. It consisted of 2,485 CSJ lectures containing 6.1 million words. The baseline language model was a word trigram interpolated with a word-class trigram based on 100 word classes. The vocabulary size of the baseline model was 30k. Interpolation weights of 0.7 and 0.3 were used for word and class models, respectively. The word class definition was trained using the incremental greedy merging algorithm [8]. This model is denoted as **GLM** (General Language Model).

Similarly to the acoustic modeling, lecture-based cluster models and utterance-based cluster models were made. The cluster definitions were trained using the same algorithm as that used for cluster-based acoustic modeling, except that bigram perplexity was used as a measure instead of acoustic likelihood. Based on the obtained definition of the clusters, lecture/utterance cluster-based models were made. Each component model was a word trigram which was trained by mixing the entire training set and the lectures/utterances in the cluster. This means that lectures/utterances in the cluster was duplicated in the training set. These cluster-based models were interpolated with the word-class trigram using the fixed interpolation weights of 0.7 and 0.3, respectively. They are denoted as **LCLM(N)** (Lecture-based Cluster Language Model) and **UCLM(N)** (Utterance-based Cluster Language Model), where N is the number of element models.

3.4. Recognition Systems

The Julius decoder [9] was used without any modification both in a baseline recognition system and decoding units of the MPD. A GRID system [10] was used for the MPD. The baseline decoding system used the speaker-independent acoustic model (GAM) and language model (GLM), whereas the MPD used the cluster-based acoustic model and language model. The number of the decoding units was a product of the number of elements of the cluster-based acoustic model and the cluster-based language model, where up to 400 decoding units were implemented. When lecture cluster-based models were used, the integrator selected recognition hypotheses throughout each lecture from one of the decoding units that maximized the total likelihood. On the other hand, when utterance cluster-based models were used, a hypothesis was selected independently for each utterance.

3.5. Unsupervised adaptation

When the response time for each utterance is not crucial, recognition can be performed in a batch, off-line mode. In such cases, batch-type unsupervised adaptation can be applied as an effective way to improve the recognition rate. In our experiments, unsupervised acoustic and language model adaptation processes were applied to the general models as well as the cluster-based models. The adaptation was conducted for each lecture based on the recognition results obtained by the baseline or the MPD-based recognition system.

For the baseline system, the acoustic model was adapted using the MLLR method and the language model was adapted using the word class-based method [11]. The word class-based language model adaptation updated word probability given a word class by maximum likelihood criteria using the recognition hypotheses. The adaptation processes for language model and acoustic models were conducted simultaneously. The adaptation for the MPD-based system was conducted in a similar way as the baseline system, in which all the element models were adapted using the MLLR and word class-based adaptation methods.

4. Experimental results

Table 2 shows the recognition results using the combination of cluster-based acoustic models and the general language model. For the lecture cluster-based acoustic modeling, the best result was obtained when the number of clusters was 10 and 20, whereas for the utterance cluster-based modeling, the lowest

Table 2: Word error rate using the cluster-based acoustic models

CAM # of clusters	Lecture cluster model	Utterance cluster model
1 (GAM)	24.9	
5	24.0	23.7
10	23.8	23.0
20	23.8	23.2

Table 3: Word error rate using the cluster-based language models

CLM # of clusters	Lecture cluster model	Utterance cluster model
1 (GLM)	24.9	
5	24.6	24.0
10	24.7	23.6
20	24.5	23.3
40	24.3	23.6

word error rate was achieved when the number of clusters was 10. The highest word error reduction rate of 7.7% was achieved under the condition of UCAM(10).

Table 3 shows the results using the combination of the general acoustic model and the cluster-based language models. The UCLM(20) achieved the highest word error rate reduction of 6.4%.

Table 4 shows the recognition results using the combination of cluster-based acoustic models and cluster-based language models. The number of decoding units increases as a product of the number of element models, and within the conditions investigated in our experiments, the lowest recognition error rate was achieved with 100 decoding units for both lecture-based clustering and utterance-based clustering. It is also observed that utterance-based cluster models achieved significantly lower word error rate than lecture-based cluster models, and the largest word error rate reduction of 11.8% was obtained by the combination of UCLM(10) and UCAM(10).

Table 4: Word error rate using the combination of cluster-based language and acoustic models

# of clusters LM x AM	# of DUs	Lecture cluster model	Utterance cluster model
1x1 (GLMxGAM)	1	24.9	
5x5	25	23.7	22.9
10x10	100	23.4	22.0
20x20	400	23.5	22.1

Figure 3 compares word error rates for each lecture using the baseline system and the MPD-based system with ACLM(10) and UCLM(10). As can be seen, error rates were reduced for all the test-set lectures. However, the error rate still varies from lecture to lecture, even using the cluster-based models.

Although only likelihood values were used for selecting hypotheses in this paper, there exist many ways to integrate the hypotheses from the DUs, including the ROVER method [12]. Table 5 shows word error rate when oracle results were used, that is, hypotheses were selected so that the error was minimized, to see the upper bound of the selection method. The word error rates in this condition are much smaller than the results shown in Table 4. Therefore, further improvement is expected by improving the selection criterion.

Figure 4 shows results when unsupervised acoustic model and language model adaptation were applied to the baseline and the MPD-based systems. The MPD-based system was based on UCAM(10) and UCLM(10). By applying the unsupervised
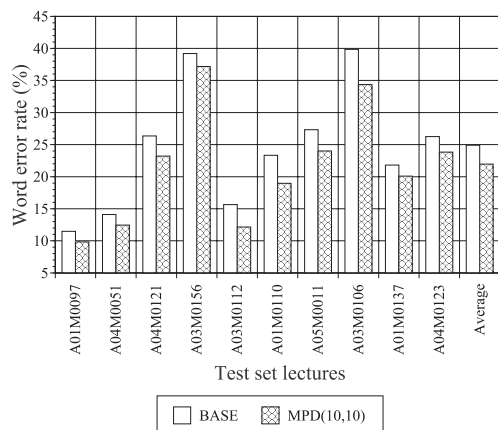
Figure 3: Word error rates for each test set lecture.

Table 5: Lower bound of word error rates obtained by hypotheses selection

# of clusters LM x AM	Lecture cluster model	Utterance cluster model
1x1 (GLMxGAM)	24.9	
5x5	23.2	18.0
10x10	22.5	16.5
20x20	22.0	15.2

adaptation to the MPD, a word error rate of 20.4% was obtained.

5. Discussion

It has been confirmed that utterance-based cluster models gave significantly lower word error rates than lecture-based cluster models. For the language modeling, this is probably because it is easier to find similar examples in the training set when the selection unit is shorter. In addition, since the recognition task is to recognize academic lectures, it is unlikely that similar stories occur multiple times. Concerning the acoustic modeling, the primary source of difference in acoustic characteristics is considered to be individuality, and this is supposed to be consistent within a lecture. Actually the difference in word error rates between lecture-based and utterance-based acoustic models is smaller than the difference for the language models, as shown in Table 2 and 3. However, the utterance-based clustering is still better than the lecture-based clustering, which probably means that voice characteristics vary from utterance to utterance and that there always exists some difference between voice characteristics of training and testing speakers.

6. Conclusion

This paper has proposed to use the Massively Parallel Decoder (MPD) consisting of a large number of decoding units and an integrator. The MPD runs using cluster-based acoustic and language models. By using parallel computers, the turnaround time hardly increases in comparison with conventional decoders using a single acoustic model and a language model. It has been found that utterance-based cluster models give significantly lower recognition error rate than lecture-based cluster models for both language and acoustic modeling. The condition of using 100 decoding units combining 10 acoustic models and 10 language models has achieved the minimum error rate which is relatively 12% lower than the result using a single de-
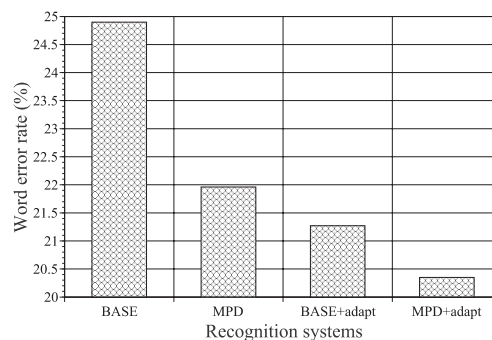

Figure 4: Word error rates with/without unsupervised acoustic as well as language model adaptation. BASE+adapt and MPD+adapt indicate results when the adaptation is applied.

coder. Future works include improving clustering algorithms and investigating integration methods for recognition hypotheses obtained from decoding units.

7. References

[1] R. Iyer and M. Ostendorf, "Modeling long distance dependence in language: Topic Mixtures vs. Dynamic Cache Models," Proc. ICSLP, Vol.1, pp. 236–239, 1996.

[2] Z. Zhang, S. Furui and K. Ohtsuki, "On-line incremental speaker adaptation with automatic speaker change detection," Proc. ICASSP, vol.2, pp. 961–964, 2000.

[3] K. Ohtsuki, K. Bessho, Y. Matsuo, S. Matsunaga, and Y. Hayashi, "Automatic indexing of multimedia content by integration of audio, spoken language, and visual information," Proc. ASRU, 2003.

[4] T. Kawahara, H. Nanjo, T. Shinozaki and S. Furui, "Benchmark test for speech recognition using the corpus of spontaneous Japanese," Proc. SSPR2003, pp. 135–138, 2003.

[5] S. Itoh, "Technical trends on Grid computing," Trans of IPSJ, Vol.44, No.6, pp. 576–580, 2003.

[6] R. R. Tummala and V. K. Madisetti, "System on Chip or System on Package," IEEE Design & Test of Computers, Vol. 16, No. 2, pp. 48–56, April 1999.

[7] S. Young, D. Kershaw, J. Odell, D. Ollason, V. Valtchev and P. Woodland, "The HTK Book, Version 2.2," Entropic Ltd, 1999.

[8] P. F. Brown, V. J. Della Pietra, P. V. deSouza, J. C. Lai and R. L.Mercer, "Class-based n-gram models of natural language," Computational Linguistics, vol.18, no.4, pp. 467–479, 1992.

[9] A. Lee, T. Kawahara and S. Doshita, "An efficient two-pass search algorithm using word trellis index," Proc. ICSLP, pp.1831–1834, 1998.

[10] http://www.gsic.titech.ac.jp/English/Publication/Event/top500.html

[11] T. Yokoyama, T. Shinozaki, K. Iwano and S. Furui, "Unsupervised class-based language model adaptation for spontaneous speech recognition," Proc ICASSP, vol.1, pp. 236–239, 2003.

[12] J. Fiscus, "A Post-Processing System to Yield Reduced Word Error Rates: Recognizer Output Voting Error Reduction," Proc. ASRU, pp. 347–352, 1997.