LOW-FREQUENCY CHARACTER CLUSTERING FOR END-TO-END ASR SYSTEM

Hitoshi Ito*, Aiko Hagiwara*, Manon Ichiki*, Takeshi Kobayakawa*, Takeshi Mishima*, Shoei Sato*, Akio Kobayashi†
* NHK (Japan Broadcasting Corp.), Japan
E-mail: {itou.h-ce, hagiwara.a-iy, ichiki.m-fq, kobayakawa.t-ko, mishima.t-iy, satou.s-gu}@nhk.or.jp
† Tsukuba University of Technology, Japan
E-mail: a-kobayashi@a.tsukuba-tech.ac.jp

***Abstract*—We developed a label-designing and restoration method for end-to-end automatic speech recognition based on connectionist temporal classification (CTC). With an end-to-end speech-recognition system including thousands of output labels such as words or characters, it is difficult to train a robust model because of data sparsity. With our proposed method, characters with less training data are estimated using the context of a language model rather than the acoustic features. Our method involves two steps. First, we train acoustic models using 70 class labels instead of thousands of low-frequency labels. Second, the class labels are restored to the original labels by using a weighted finite state transducer and n-gram language model. We applied the proposed method to a Japanese end-to-end automatic speech-recognition system including labels of over 3,000 characters. Experimental results indicate that the word error rate relatively improved with our method by a maximum of 15.5% compared with a conventional CTC-based method and is comparable to state-of-the-art hybrid DNN methods.**

***Index Terms*—end-to-end ASR, acoustic modeling, connectionist temporal classification, long short-term memory**

I. Introduction

Automatic speech recognition (ASR) technology has been improved using hidden Markov models (HMMs) and deep neural networks (DNNs). The hybrid HMM-DNN system uses the states of HMMs to handle the difference in the sequence length between acoustic features and output labels. Typically, three succeeding states of these HMMs are associated with phonemes. The phonemes are mapped to words using a pronunciation dictionary. On the other hand, the acoustic model of the end-to-end ASR system can train the relationship between acoustic features and characters or words directly. In contrast to a state transition model in HMMs, connectionist temporal classification (CTC) [1] absorbs the difference between the sequence of acoustic features and the output labels by using a blank label. CTC typically uses long short-term memory (LSTM) [2] or bi-directional LSTM (BLSTM) [3], [4]. The performance of end-to-end ASR systems are comparable with that of HMM-DNN hybrid systems using CTC [5]–[15].

However, it is difficult to train end-to-end ASR systems handling thousands of output labels. If we use an acoustic model that involves basic units such as words or thousands of unique characters, we have to solve the data-sparsity problem. For example, we have to train 3,000 and 6,000 output labels for Japanese and Chinese, respectively, while we only have to train at most about 100 output labels for English. When the number of output labels is large, the number of training samples of many labels is insufficient. To solve this problem, we believe it is necessary to increase the number of training samples for each label by redesigning the label set.

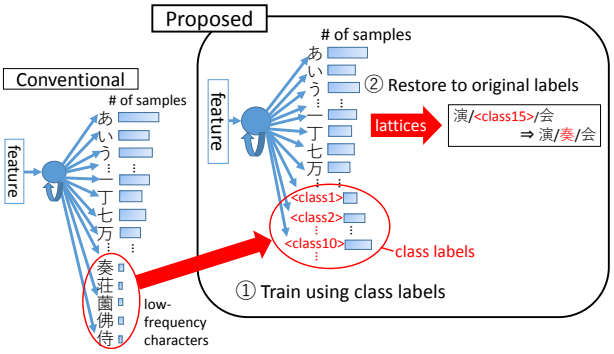

Fig. 1. Overview of model using low-frequency character clustering

Studies on redefining the output labels have been conducted [13], [16]. These studies aimed at examining appropriate character delimiters for robustly representing acoustic features. However, since they increased the number of output labels, the data-sparsity problem was not solved. To increase the number of training samples for each label, conventional systems typically use phoneme or sub-phoneme units. The system developed by Audhkhasi et al. [14] uses a model initialized using a phoneme-based model to train a word-based model. The system developed by Kanda et al. [15] uses syllables as output labels. However, few studies have been conducted for solving the data-sparsity problem without using phonemes or sub-phoneme units.

In this paper, we propose a label-designing and restoration method for end-to-end speech recognition using class-based labels for acoustic modeling in CTC. An overview of the model using the proposed method is shown in Fig. 1. Our system consists of two models. One is a character-based acoustic model including class labels. This acoustic model replaces thousands of rare labels with 70 class labels. This clustering is expected to improve robustness similar to state

tying of HMMs [17]. Unlike other syllable-based methods, we apply syllable-based approximate clustering for only low-frequency characters. The other is a language model including an additional weighted finite-state transducer (WFST) [18] that restores the class labels to their original labels. The WFST expands the lattice including class labels into a lattice of only original labels using context.

The proposed method correctly estimates these low-frequency characters by restoration using the above language model. Our experiments involving Japanese TV program showed that a Japanese-character-based system using our method is superior to a current end-to-end ASR system [9] and that the word error rate (WER) of our method is on par with hybrid DNN methods.

II. Connectionist Temporal Classification (CTC)

CTC is an approach to map the acoustic feature sequence directly to symbols such as characters and words. The posteriori of the output string $c = \{c_1, c_2, ..., c_T\}$ for input sequence X is expressed by the following equation.

$$\mathrm{P}(c|X) = \prod_{t=1}^{T} u_t^{c_t}, \tag{1}$$

where $u_t^{c_t}$ is the output score estimated by the probability of c_t at time t. CTC introduces a blank symbol ø to absorb the difference in sequence length among inputs and outputs. The symbol ø is estimated to fill frames between characters. For example, "AAøBøCC" and "AøBBBøCø" are mapped to the same symbols sequence "ABC". Consequently, the posterior probability of the output label series z is expressed by the following equation.

$$\mathrm{P}(z|X) = \sum_{c \in \Phi(z)} \mathrm{P}(c|X), \tag{2}$$

where $\Phi(z)$ is all series whose output label sequence is z, the element c is mapped to the same z. CTC trains the connection weight W to minimizes the error function $\mathrm{E}(W)$.

$$\mathrm{E}(W) = -\sum_{n \in N} \log \mathrm{P}(z_n|X_n), \tag{3}$$

where (X_n, z_n) is a pair of input and output series for given sentence n, which is the element of the training data N.

III. Label-designing and restoration method for end-to-end ASR system

Our method converts low-frequency characters to class labels and restores them to original characters using context. In this paper, the acoustic model is trained using a newly designed label set instead of labels with a small number of training samples. The designed label set, therefore, consists of labels with sufficient training samples and class labels. Since the number of labels in the acoustic model is reduced using class labels, the number of training samples per label increases. When decoding, class labels are restored from a lattice hypothesis including the class labels to original labels using a WFST. Our WFST contains the mapping from characters including class labels to words.

A. Label designing for low-frequency characters

Low-frequency characters are classified into classes based on syllable-based expressions.

However, there are two problems when clustering kanji (Chinese-based characters used in Japanese) based on syllables. The first problem is that one kanji can contain a sequence of consecutive syllables. For example, the kanji '寿' consists of four connected syllables, "ko" "to" "bu" "ki". In this paper, for the sake of simplicity, it is divided into classes for each first syllable of each character. The second problem is that one Japanese kanji can have many readings. For example, the kanji '生' has more than ten completely different readings such as "nama", "sei", and "ki". In this paper, when clustering kanji with multiple readings, we use the typical reading of that character. The number of classes is 70, which is the total number of syllables.

B. Restoration to original labels

The acoustic model adopting the class model outputs tokens including class labels. To output a recognition hypothesis, it is necessary to restore the class labels to the original characters. We restore class labels to low-frequency characters by using a trusted language model. With only the training data of the acoustic model, it is difficult to learn all labels robustly. On the other hand, a highly reliable language model can be trained because the training data of the language model are easier to collect than those of the acoustic model. We believe that it is more reliable to estimate low-frequency characters by using a language model instead of a NN.

We estimate low-frequency characters by using WFST decoding that extends Miao et al.'s method [9]. A WFST is a converter that transitions information by writing pairs of input and output signals and their weights. In this study, we decoded by synthesizing three transducers, i.e., from CTC label to character (T), character to word (L), and word to sentence (G). Transducer composition S is expressed as

$$S = T \circ \min(\det(L \circ G)) \tag{4}$$

With our proposed method, we restore the original label from the class label by transducer L, which converts the token sequence including the class labels estimated by transducer T into a word containing the original character. The FST of the word "演奏会" ("concert") when training by assigning the kanji '奏' to class label "<class15>" is shown in Fig. 2.

The original label is determined by the posterior of the class label and the n-gram described by transducer G.

IV. Experiments

A. Setup

We used NHK's informative TV show "Hirumae Hotto" consisting of about 32 k words as evaluation data. The data consists of clean utterances, including read speech of news manuscripts, and noisy utterances, including background music or field noise. The noisy utterances contain spontaneous speech. For our end-to-end ASR experiments, the baseline

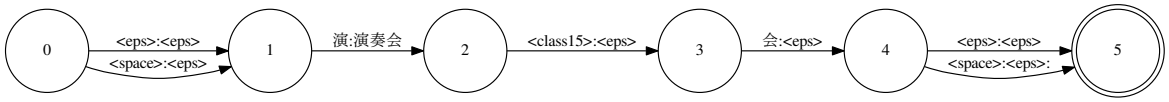

Fig. 2. Example of restoration using FST

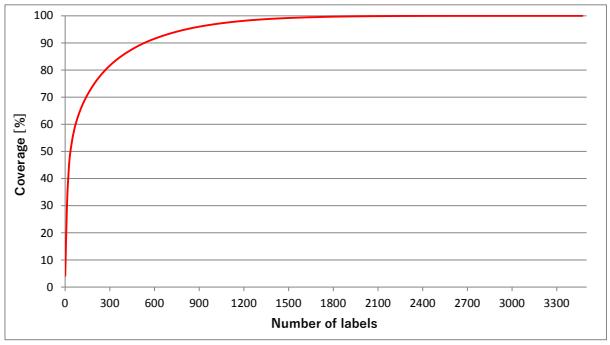

Fig. 3. Cumulative frequency distribution of labels in training data

used the EESEN [9] framework based on the Kaldi toolkit [19], which we modified to enable Japanese-character output. We trained a CTC-based BLSTM for the acoustic model using NHK broadcast programs and their closed-captions as training data. The CTC model was trained using 712 hours of training data on 4-layer BLSTM. The BLSTM contains 320 memory cells at each layer for forward and backward. We adopted a "$newbob$" annealing schedule. This scheduling reduces the learning rate by half if the cross-varidation frame accuracy did not increase by more than 0.5 compared with the previous epoch. The initial value of the learning rate was set to 5.0×10^{-5}. We applied low-rank matrix decomposition [20], [21] to affine transformation layer at the output end of the network. The NN that outputs Japanese characters has many parameters due to its large number of labels. This matrix decomposition is aimed at reducing learning time. We replaced the affine transform layer full-rank matrix of $640 \times V$, with two matrices, one of 640×320 and another of $320 \times V$, where V is the output label size. We used 40-dimentional filter-bank features together with their first and second-order delta derivatives as input acoustic features. We trained using a WFST language model as a 3-gram. The language model was estimated from a total of 620 million words in the NHK news manuscripts and closed-captions with a 200-k-word vocabulary. The output labels of the network consist of 1,500 high-frequency characters and 70 class labels. In our training data, the top 1,500 high-frequency labels covered 99% of the data (Fig. 3). The remaining 1,977 characters were assigned to 70 class labels.

We used the following model as the baseline:

baseline: The model based on Miao et al.'s method [9] trained with the above parameters. The class labels are not used.

In order to compare with the clustering criterion in Sec.III-A, we proposed two additional clustering criteria. We compared the following three clustering models with our proposed method:

class (all): The model trained with our method that assigns all the remaining characters to one class label.

class (random): The model trained with our method that randomly assigns the selected labels to 70 class labels.

class (reading): The model trained with our method that assigns the selected labels to 70 class labels based on their reading, as mentioned in Sec.III-A.

We also compared two conventional models based on NNs:

HMM-DNN: The model trained with HMM-DNN cross-entropy training based on Moriya et al.'s method [22].

TDNN: The model trained with a time-delay neural network (TDNN) [23] based on Kaldi recipe.

We used the default parameters in these conventional models. The training data of the acoustic and language models were the same as those of the baseline.

B. Results

1) Difference in clustering methods:

Table I compares the WERs (%) of the three clustering models trained with the proposed method with their number of labels in the"#Labels" column. The results of HMM-DNN and TDNN are also shown in the same manner. The results indicate that these class models improved the WER, regardless of clustering criteria. This is because smoothing of the output score of the NN by introducing class labels. By smoothing the score, correct hypotheses remain as candidates for rescore. If a hypothesis remains as the output of an NN, we can output it using the rescore of the language model. There was significant improvement in the "class (reading)" model. This clustering model exceeded the WERs of the HMM-DNN and TDNN models. Performance improved using 70 classes compared to using one class. This is probably because the smaller number of characters belonging to each class, the higher the prediction performance of the language model. However, in the case of the same class number, there was no

TABLE I
PERFORMANCE OF VARIOUS CLUSTERING MODELS WITH OUR PROPOSED METHOD, AND COMPARISON WITH RESULTS PRESENTED IN PREVIOUS WORK

Model	# of Labels	WER
baseline	3,477	14.2
class (all)	1,501	13.1
class (random)	1,570	12.2
class (reading)	1,570	12.0
HMM-DNN	-	12.8
TDNN	-	12.7

TABLE II
NUMBER OF WORDS THAT CONTAIN LOW-FREQUENCY CHARACTERS IN RECOGNITION HYPOTHESES

Model	#Words
baseline	1,359
class (reading)	662
reference	665

significant difference in WER among the clustering models. This is thought due to approximation of clustering. In the "class (reading)" model, since clustering is based on the first syllable of the representative reading of each character, the variance in the acoustic-feature quantity for each class is not small.

2) Difference in recognition hypotheses:

Next, we compared the speech-recognition hypotheses. Table II shows the number of words that contain low-frequency characters in the recognition hypotheses. In the baseline, the number of low-frequency characters included in the recognition hypotheses was 694 more than the reference. The baseline model unnecessarily output words containing low-frequency characters. From this result, we assume that the acoustic features of low-frequency characters are not correctly learned.

Tables III and IV list examples of the recognition hypotheses of the baseline and class (reading) models. Bold letters indicate low-frequency characters.

As shown in Table III, unnecessary low-frequency characters were inserted in the baseline hypothesis. The noise portion after the utterance was erroneously recognized as words including a low-frequency character 'え'. Such insertion errors of low-frequency characters in the non-speech part were observed in many places. We assume this is one of the causes of WER degradation with the baseline. With the class (reading) model trained with our proposed method, on the other hand, this unnecessary output was suppressed. This indicates that the acoustic features of low-frequency characters are properly learned. When estimating the recognition hypothesis somewhere in the non-speech segment, the acoustic feature of noise is erroneously estimated as a low-frequency character that was not well learned.

As shown in Table IV, a low-frequency character '瓶' was correctly restored with the class (reading) model trained with our proposed method. By applying a language model, we correctly recognized the characters with insufficient learning data. This indicates that labels with which it is difficult to learn acoustic features can be estimated correctly using a language model. From these results, the proposed method not only suppresses unnecessary generation of low-frequency characters but also restores low-frequency characters to correct positions depending on contexts.

TABLE III
EXAMPLE1: RECOGNITION HYPOTHESES

Model	Hypotheses
reference	祭り と 呼ば れ て い ます
baseline	祭り と 呼ば れ て い ます ねぇ ねぇ ねぇ
class (reading)	祭り と 呼ば れ て い ます

TABLE IV
EXAMPLE2: RECOGNITION HYPOTHESES

Model	Hypotheses
reference	そして 花瓶 に 生け られ た
baseline	そして 鼻 に 抜け られ た
class (reading)	そして 花瓶 に 生け られ た

V. CONCLUSION

We proposed a label-designing and restoration method for tasks with many output labels in end-to-end speech recognition. Although CTC-based end-to-end ASR systems handling thousands of output labels cause data-sparsity problems, we solved this problem by replacing low-frequency labels with class labels. Our method reduces the occurrence of two types of misrecognition errors insertion errors of low-frequency characters and the other is substitution errors of low-frequency characters, which are suppressed using a language model. The trained models are considered to be smoothed by introducing class labels. We experimentally showed that the discriminative ability degraded by smoothing is restored using the WFST of the proposed method. It is considered that the paradigm of the class-label smoothing is similar to the state tying of an HMMs. Experimental results indicate that our method is on a par with conventional state-based methods. Our method improves the WER even if the criterion of clustering is not strict on pronunciation. The best criterion of our method was syllable-based clustering. However, since approximation is used to assign each kanji to a syllable-based class, a large performance difference from random clustering was not observed. Future work will involve data-oriented label designing for more appropriate clustering.

REFERENCES

[1] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *Proc. ICML*. ACM, 2006, pp. 369–376.

[2] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[3] M. Schuster and K. K. Paliwal, "Bidirectional recurrent neural networks," *IEEE Transactions on Signal Processing*, vol. 45, no. 11, pp. 2673–2681, 1997.

[4] A. Zeyer, R. Schlüter, and H. Ney, "Towards online-recognition with deep bidirectional LSTM acoustic models," in *Proc. Interspeech*, 2016, pp. 3424–3428.

[5] A. Graves, A.-R. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," in *Proc. ICASSP*. IEEE, 2013, pp. 6645–6649.

[6] A. Graves and N. Jaitly, "Towards end-to-end speech recognition with recurrent neural networks." in *Proc. ICML*, vol. 14, 2014, pp. 1764–1772.

[7] A. Hannun, C. Case, J. Casper, B. Catanzaro, G. Diamos, E. Elsen, R. Prenger, S. Satheesh, S. Sengupta, A. Coates *et al.*, "Deep speech: Scaling up end-to-end speech recognition," *arXiv preprint arXiv:1412.5567*, 2014.

[8] D. Amodei, R. Anubhai, E. Battenberg, C. Case, J. Casper, B. Catanzaro, J. Chen, M. Chrzanowski, A. Coates, G. Diamos *et al.*, "Deep speech 2: End-to-end speech recognition in English and Mandarin," *arXiv preprint arXiv:1512.02595*, 2015.

[9] Y. Miao, M. Gowayyed, and F. Metze, "EESEN: End-to-end speech recognition using deep RNN models and WFST-based decoding," in *Proc. ASRU*. IEEE, 2015, pp. 167–174.

[10] A. L. Maas, Z. Xie, D. Jurafsky, and A. Y. Ng, "Lexicon-free conversational speech recognition with neural networks." in *Proc. HLT-NAACL*, 2015, pp. 345–354.

[11] H. Sak, A. Senior, K. Rao, O. Irsoy, A. Graves, F. Beaufays, and J. Schalkwyk, "Learning acoustic frame labeling for speech recognition with recurrent neural networks," in *Proc. ICASSP*. IEEE, 2015, pp. 4280–4284.

[12] T. Hori, S. Watanabe, Y. Zhang, and W. Chan, "Advances in joint CTC-attention based end-to-end speech recognition with a deep CNN encoder and RNN-LM," in *Proc. Interspeech*, 2017, pp. 949–953.

[13] H. Liu, Z. Zhu, X. Li, and S. Satheesh, "Gram-CTC: Automatic unit selection and target decomposition for sequence labelling," *arXiv preprint arXiv:1703.00096*, 2017.

[14] K. Audhkhasi, B. Ramabhadran, G. Saon, M. Picheny, and D. Nahamoo, "Direct acoustics-to-word models for English conversational speech recognition," *arXiv preprint arXiv:1703.07754*, 2017.

[15] N. Kanda, X. Lu, and H. Kawai, "Maximum a posteriori based decoding for CTC acoustic models," in *Proc. Interspeech*, 2016, pp. 1868–1872.

[16] W. Chan, Y. Zhang, Q. Le, and N. Jaitly, "Latent sequence decompositions," *arXiv preprint arXiv:1610.03035*, 2016.

[17] S. J. Young, J. J. Odell, and P. C. Woodland, "Tree-based state tying for high accuracy acoustic modelling," in *Proceedings of the workshop on Human Language Technology*. Association for Computational Linguistics, 1994, pp. 307–312.

[18] M. Mohri, F. Pereira, and M. Riley, "Weighted finite-state transducers in speech recognition," *Computer Speech & Language*, vol. 16, no. 1, pp. 69–88, 2002.

[19] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz *et al.*, "The Kaldi speech recognition toolkit," in *Proc. ASRU*, no. EPFL-CONF-192584. IEEE Signal Processing Society, 2011.

[20] T. N. Sainath, B. Kingsbury, V. Sindhwani, E. Arisoy, and B. Ramabhadran, "Low-rank matrix factorization for deep neural network training with high-dimensional output targets," in *Proc. ICASSP*. IEEE, 2013, pp. 6655–6659.

[21] H. Ito, A. Hagiwara, M. Ichiki, T. Mishima, S. Sato, and A. Kobayashi, "End-to-end speech recognition for languages with ideographic characters," in *Proc. APSIPA*, 2017.

[22] T. Moriya, T. Shinozaki, and S. Watanabe, "Kaldi recipe for Japanese spontaneous speech recognition and its evaluation," in *Autumn Meeting of ASJ*, no. 3-Q-7, 2015.

[23] V. Peddinti, D. Povey, and S. Khudanpur, "A time delay neural network architecture for efficient modeling of long temporal contexts," in *Proc. Interspeech*, 2015.