The NICT ASR System for IWSLT2012

Hitoshi Yamamoto, Youzheng Wu, Chien-Lin Huang, Xugang Lu, Paul R. Dixon, Shigeki Matsuda, Chiori Hori, Hideki Kashioka

Spoken Language Communication Laboratory,
National Institute of Information and Communication Technology,
Kyoto, Japan

hitoshi.yamamoto@nict.go.jp

Abstract

This paper describes our automatic speech recognition (ASR) system for the IWSLT 2012 evaluation campaign. The target data of the campaign is selected from the TED talks, a collection of public speeches on a variety of topics spoken in English. Our ASR system is based on weighted finite-state transducers and exploits an combination of acoustic models for spontaneous speech, language models based on n-gram and factored recurrent neural network trained with effectively selected corpora, and unsupervised topic adaptation framework utilizing ASR results. Accordingly, the system achieved 10.6% and 12.0% word error rate for the tst2011 and tst2012 evaluation set, respectively.

1. Introduction

This paper describes our automatic speech recognition (ASR) system for the IWSLT 2012 evaluation campaign.

The target speech data of the ASR track of the campaign is selected from TED talks, a collection of short presentations to an audience spoken in English. These talks are generally in spontaneous speaking style, which touch on a variety of topics related to Technology, Entertainment and Design (TED). Main challenges of the track are clean transcription of spontaneous speech, detection and removal of non-words, and talk style and topic adaptation [1].

An overview of our ASR system is depicted in Figure 1. The core decoder of the system is based on weighted finite-state transducers (WFSTs). It exploits two types of state-of-the-art acoustic models (AMs) of spontaneous speech which are integrated in lattice level. Here, n-gram language models (LMs) are trained with in-domain and effectively selected out-of-domain corpora. Then, it employs recurrent neural network (RNN) based LMs newly extended to incorporate additional linguistic features. Finally, it utilizes ASR results to adapt LMs to talk style and topic.

This paper is organized as follows. Section 2 explains the training data and procedure of AMs in the system. Section 3 presents an overview of the data and technique used to build and adapt our LMs. Section 4 describes decoding strategy and experimental results.

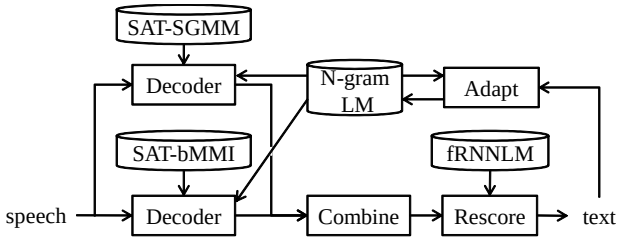

Figure 1: *Overview of the NICT ASR system for IWSLT2012.*

2. Acoustic Modeling

2.1. Training Corpus

To train AMs suitable for TED talks, we crawled movies and subtitles of talks published prior to 2011 from the TED website[1]. The collected 777 talks contain 204 hours audio and $1.8M$ words, excluding 19 talks of the development set (dev2010, tst2010).

For each talk, the subtitle is aligned to the audio of the movie because it doesn't contain accurate time stamps of speech segments for training phoneme-level acoustic models. We utilize SailAlign [2] to extract text-aligned speech segments from the audio data. As shown in Figure 2, it iterates two steps, (a) text-based alignment of ASR results and transcriptions and (b) ASR model adaptation using text-aligned speech segments. Here it runs with its basic setting, using HTK and AM trained on WSJ. After two iterations, 170 hours of text-aligned speech segments (with $1.6M$ words) are defined as AM training corpus.

2.2. Training Procedure

The acoustic feature vector has 40 dimensions. We first extract 13 static MFCCs including zeroth order for each frame (25ms width and 10ms shift) and normalize them with cepstrum mean normalization for each talk. Then, for each frame, we concatenate MFCCs of 9 adjacent frames (4 on each side of the current frame) and apply transformation matrix based on linear discriminant analysis (LDA) and maxi-

[1] http://www.ted.com/

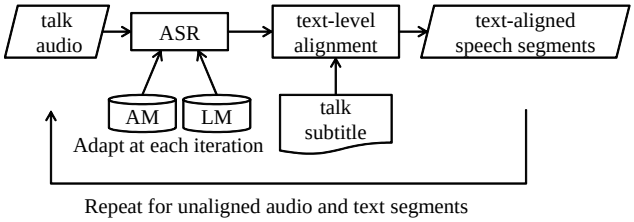

Figure 2: *Adaptive and iterative scheme of SailAlign [2].*

mum likelihood linear transformation (MLLT) to reduce its dimension to 40. In addition, we apply feature space MLLR for speaker adaptive training for each talk, assuming that one talk includes one speaker.

The acoustic models are cross-word triphone HMMs of which units are derived from 39 phonemes. Each phoneme is classified by its position in word (4 classes: begin, end, singleton and the others) and each vowel is further distinguished by its accent mark (3 classes: first, second and the others).

Three types of acoustic models are developed with the Kaldi speech recognition toolkit [3] revision 941. We first train HMMs with GMM output probability. This model totally include 6.7K states and 80K Gaussians trained with ML estimation (SAT-ML). Then we increase the number of Gaussian of it to $240K$ (other parts are not changed) and train them with boosted MMI criterion (SAT-bMMI). We also build HMMs with subspace GMM output probability. This model consists of $9.1K$ states, which is transformed from the SAT-ML model (SAT-SGMM).

3. Language Modeling

3.1. Training Corpus

The IWSLT evaluation campaign defines a closed set of publicly available English texts as training data of LM. We use the in-domain corpus (transcription of TED talks) and parts of the out-of-domain corpora (English Gigaword Fifth Edition and News Commentary v7) and pre-process the data as follows: (1) converting non-standard words (such as CO2 or 95%) to their pronunciations (CO two, ninety five percent) using a non-standard-word expansion tool[2] [4], and (2) removing duplicated sentences. The statistics of the preprocessed corpora are shown in Table 1.

The lexicon consists of the CMU Pronouncing Dictionary[3] v.0.7a. In addition, we extract new words (not included in the CMU dictionary) from the preprocessed in-domain corpora and generate their pronunciations with a WFST-based grapheme-to-phoneme (G2P) technique [5]. The extended lexicon contains $156.3K$ pronunciation entries of $133.3K$ words which are used as the LM vocabulary with an OOV rate of 0.8% on the dev2010 data set.

[2] http://festvox.org/nsw

[3] http://www.speech.cs.cmu.edu/cgi-bin/cmudict

Table 1: *Statistics of English LM training corpora*

	Corpus	#sentences	#words
in-domain	TED Talks	$142K$	2,402K
out-of-domain	News Commentary	$212K$	4,566K
	English Gigaword	$123M$	2,722M

3.2. Domain adaptation

The large out-of-domain corpora likely includes sentences that are so unlike the domain of the TED talks. LM trained on these unlike sentences is probably harmful. Therefore, we adopt domain adaptation by selecting only a portion of the out-of-domain corpus instead of using the whole.

We employ cross-entropy difference metric for domain adaptation, which biases towards sentences that are both like the in-domain corpus and unlike the average of the out-of-domain corpus [6]. Each sentence s of the out-of-domain corpus is scored as follows,

$$H_I(s) - H_O(s), \tag{1}$$

where $H_I(s)$ and $H_O(s)$ represent cross-entropy scores according to LM_I trained on the in-domain corpus, and LM_O trained on a subset sentences randomly selected from the out-of-domain corpus. Here, LM_I and LM_O are similar size. Then the lowest-scoring sentences are selected as a subset of out-of-domain corpus.

3.3. N-gram LM

For the in-domain and the selected out-of-domain corpora, modified Kneser-Ney smoothed n-gram LMs (n=3,4) are constructed using SRILM [7]. They are interpolated to form a baseline of n-gram LMs by optimizing the perplexity of the development data set. To apply the domain adaptation, we empirically select $1/4$ of the out-of-domain corpus with $30M$ sentences and $559M$ words using Eq. (1).

3.4. Factored RNNLM

Recently, recurrent neural network (RNN) based LMs [8] become an increasingly popular choice for LVCSR tasks due to consistent improvements. In our system, we employ a factored RNNLM that exploits additional linguistic information, including morphological, syntactic, or semantic. This novel approach was proposed in our previous studies [9].

In the official run, our factored RNNLM uses two types of features, word surface and part-of-speech tagged by GENIA Tagger[4]. Other types of linguistic features are investigated in [10]. We set the number of hidden neurons in the hidden layer and the number of classes in the output layer to 480 and 300.

Since it is very time consuming to train factored RNNLM on large data, we select a subset sentences of the out-of-

[4] http://www.nactem.ac.uk/tsujii/software.html

Table 2: *Word error rate (WER, %) of the development sets and test sets. The results of primary run in our submission are represented by italic characters.*

Step	dev2010	tst2010	tst2011	tst2012
1a. Boosted MMI	16.7	14.5	12.3	13.9
1b. Subspace GMM	17.3	14.9	12.9	14.2
2. System combination	16.4	13.8	12.0	13.3
3. Factored RNNLM	15.3	13.1	10.9	12.1
4. Topic adaptation	*15.0*	*12.8*	**10.6**	**12.0**
4a. Post-processing	14.8	12.6	*10.9*	*12.1*
4b. Our decoder	—	—	**10.6**	**12.0**

domain corpus with Eq. (1) and uses it together with the in-domain corpus for training. Finally, the training data of factored RNNLM contains 1,127K sentences with 30M words.

3.5. Topic adaptation

The TED talks in the IWSLT test sets touch on various topics without adhering to a single genre. To model each test set better, we utilize first-pass recognition hypothesis for topic adaptation of n-gram LMs. A problem here is that recognition hypothesis includes errors that limits the adaptation performance. To avoid negative impact of the errors in the first-pass result, we propose a similar metric to Eq. (1), which takes into account the recognition hypothesis and randomly selected sentences of out-of-domain corpus. Our adaption can be expressed as,

$$H_{ASR}(s) - H_O(s). \tag{2}$$

For each test set, we rank sentences of the out-of-domain according to Eq. (2), select $1/8$ of sentences with the lowest scores, build an adapted n-gram LM based on the selected sentences, interpolate the adapted LM with the in-domain LM by optimizing the perplexity of the development set. Here, the lexicon is extended to include new words appearing more than 10 times in the selected sentences.

4. Decoding system

4.1. Decoding system

The procedure of our ASR system depicted in Figure 1 is divided into four steps as follows:

1. Decode input speech using two sets of models,
2. Combine lattices output from the decoders,
3. Rescore n-best with factored RNNLM,
4. Adapt LMs and run through the steps above again.

First, we use WFST-based decoder to create lattice for input speech. In the submitted system, we employ decoder of the Kaldi toolkit for 3-gram decoding and 4-gram lattice rescoring. Here, two types of AMs described in Section 2.2, (a) SAT-bMMI and (b) SAT-SGMM, are employed individually, with n-gram LMs described in Section 3.3. This step produce two lattices l_a and l_b corresponding to the two AMs.

Then, the two lattices are combined using WFST compose operation as follows:

$$l_c = \mathrm{compose}(\mathrm{scale}(w, l_a),\ \mathrm{scale}(1-w, l_b)), \tag{3}$$

where scale is used to scale transition costs of WFST with the given weight w (set to 0.5) and $\mathrm{compose}$ is an operation to compute the composition of the two input WFSTs. When the resulting lattice l_c is empty, l_a is output instead of it. Note that the $\mathrm{project}$ operation is applied to l_b before the $\mathrm{compose}$ to map its output symbols on transitions to input side.

In the third step, factored RNNLM based rescoring is applied to n-best list extracted from the lattice l_c (n=100). The LM score of input i-th sentence s_i in the n-best is calculated as an interpolation of two kinds of LMs,

$$P(s_i) = \gamma \times P_{fRNN}(s_i) + (1-\gamma) \times P_{4g}(s_i), \tag{4}$$

where γ is a weighting factor (set to 0.5), $P_{fRNN}()$ and $P_{4g}()$ stand for scores based on factored RNN and 4-gram LMs, respectively. Then the 1-best sentence is obtained from the n-best scored by Eq. (4).

In the final step, n-gram LMs and lexicon are adapted to each test set, using the topic adaptation technique described in Section 3.5. Using 1-best results of the previous step, training data is newly selected from out-of-domain corpora with Eq. (2). Then the system run through the steps 1 to 3 again as a second pass decoding with the adapted LMs. Note that the AMs and factored RNNLM are not updated here.

4.2. Evaluation Results

Table 2 shows performance of our ASR system on transcribing the development sets, dev2010 and tst2010, and the test sets, tst2011 and tst2012. Word error rates (WERs) were decreased by combining two lattices derived from different types of AMs (Step 2). With respect to LMs, rescoring using factored RNNLM significantly contributed to achieve better performance (Step 3) and topic adaptation based on dynamic data selection also showed improvement (Step 4). These results would appear that each of technique employed in our system has a particular ability to improve ASR performance, although there are some exceptional cases in talk-level as shown in Figure 3.

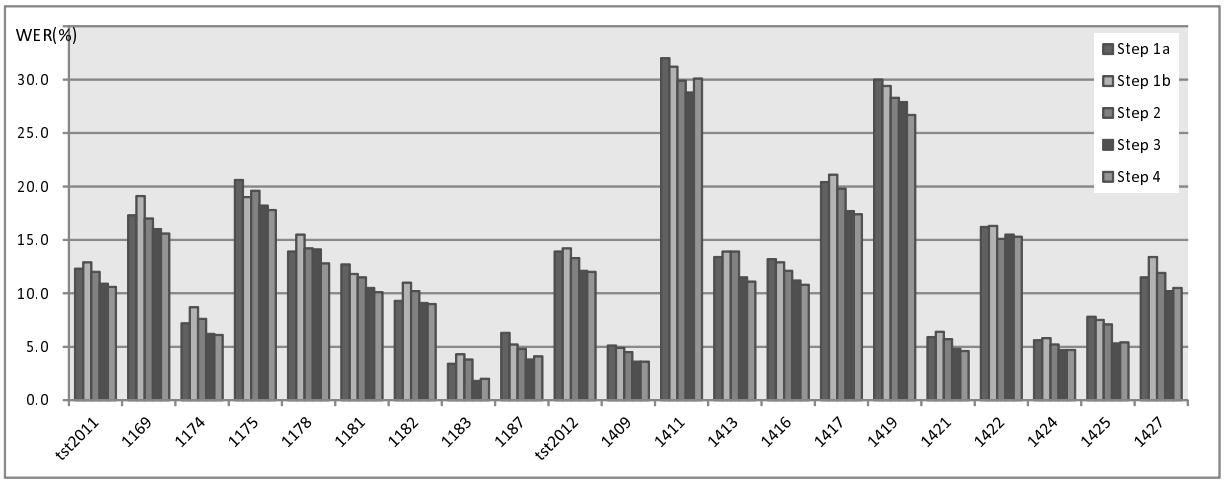

Figure 3: *Talk-level WERs of the tst2011 and tst2012.*

Note that the ASR results of the Step 4 are post-processed in our test submission (Step 4a). This step shrinks repetitions of one word or two words in word sequence. Though it helps to decrease WER of the development sets, it results in higher WER for the test sets.

Table 2 also shows the performance of our system when it utilizes our own WFST-based decoder (a variant of [11]) which can compose LMs on-the-fly during decoding time (Step 4b). The decoding process in Step 1 runs on-the-fly 4-gram decoding instead of the 4-gram rescoring after the 3-gram decoding, and also allowed for a more efficient graph building scheme. It achieved a reduction in computing time and memory usage when composing the WFSTs and running the decoder. Compared to the submitted system, it used 3% time and 26% memory in composing and 48% time and 46% memory in decoding.

5. Summary

In this paper, we describe our ASR system for the IWSLT 2012 evaluation campaign. The WFST-based system including system combination in terms of state-of-the-art AMs, factored RNNLM based rescoring, and unsupervised topic adaptation with dynamic data selection indicated an improvement in WER on transcribing the TED talks.

6. Acknowledgements

The authors would like to thank Mr. K. Abe for discussions on developing the ASR system and Dr. J. R. Novak for providing G2P toolkit.

7. References

[1] M. Federico, M. Cettolo, L. Bentivogli, M. Paul and S. Stüker, "Overview of the IWSLT 2012 Evaluation Campaign," in *Proc. of IWSLT*, 2012.

[2] A. Katsamanis, *et al.*, "SailAlign: Robust long speech-text alignment," in *Proc. of Workshop on New Tools and Methods for Very-Large Scale Phonetics Research*, 2011.

[3] D. Povey, *et al.*, "The Kaldi Speech Recognition Toolkit," in *Proc. of Workshop on Automatic Speech Recognition and Understanding*, 2011.

[4] R. Sproat, *et al.*, "Normalization of non-standard words," *Computer Speech and Language*, Vol. 15, pp. 287–333, 2001.

[5] J. R. Novak, *et al.*, "Improving WFST-based G2P Conversion with Alignment Constraints and RNNLM N-best Rescoring," in *Proc. of Interspeech*, 2012.

[6] R. Moore and W. Levis, "Intelligent selection of language model training data," in *Proc. of ACL*, 2010.

[7] A. Stolcke, *et al.*, "SRILM at Sixteen: Update and Outlook," in *Proc. of Workshop on Automatic Speech Recognition and Understanding*, 2011.

[8] T. Mikolov, *et al.*, "Recurrent neural network based language model," in *Proc. of Interspeech*, 2010.

[9] Y. Wu, *et al.*, "Factored Language Model based on Recurrent Neural Network," in *Proc. of COLING*, 2012.

[10] Y. Wu, *et al.*, "Factored Recurrent Neural Network Language Model in TED Lecture Transcription," in *Proc. of IWSLT*, 2012.

[11] P. R. Dixon, *et al.*, "A Comparison of Dynamic WFST Decoding Approaches," in *Proc. of ICASSP*, 2012.