The NICT ASR System for IWSLT 2014

Peng Shen, Xugang Lu, Xinhui Hu, Naoyuki Kanda, Masahiro Saiko, Chiori Hori

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

peng.shen@nict.go.jp

Abstract

This paper describes our automatic speech recognition system for IWSLT2014 evaluation campaign. The system is based on weighted finite-state transducers and a combination of multiple subsystems which consists of four types of acoustic feature sets, four types of acoustic models, and N-gram and recurrent neural network language models. Compared with our system used in last year, we added additional subsystems based on deep neural network modeling on filter bank feature and convolutional deep neural network modeling on filter bank feature with tonal features. In addition, modifications and improvements on automatic acoustic segmentation and deep neural network speaker adaptation were applied. Compared with our last year's system on speech recognition experiments, our new system achieved 21.5% relative improvement on word error rate on the 2013 English test data set.

1. Introduction

TED talks are presentations to audience with wide topics related to Technology, Entertainment and Design (TED) in spontaneous speaking style [1]. Automatically transcribing TED talks with automatic speech recognition (ASR) technique is still a challenging task. The difficulties are due to the large variations of TED speech caused by many factors, for example, variations caused by disfluency, emotion, noise distortions, as well as variations caused by accent and ages of speakers. In this paper, we describe our ASR system for the English TED ASR track of the 2014 IWSLT evaluation campaign.

The system is a further development of our 2012 and 2013 ASR systems which utilized lots of state of the art technologies [2, 3]. An overview of our ASR system is depicted in Figure 1. In this figure, there are several processing blocks. The test TED talks were provided without any acoustic segmentation information. For convenience of processing and decoding, an automatic acoustic segmentation was first applied. Based on the segmentation, acoustic features were extracted. Next, decoding was applied on four types of acoustic models to produce decoding lattices, and rescoring was used on the N-best lists generated by the lattices. Based on the N-best lists, a ROVER processing was used to get the first pass ROVER result. Based on the first pass ROVER result, the language model adaptation and acoustic model adaptation were done. Then decoding and rescoring were done again with the adapted LM and acoustic models. Furthermore, the second pass ROVER was conducted. The adaptation, decoding, rescoring, and ROVER were done for several rounds.

Compared with the system we used in last year, new contributions are (1) refined acoustic segmentation algorithm; (2) deep neural network (DNN) acoustic model trained based on new types of acoustic features; (3) convolutional DNN (CNN-DNN) acoustic model trained based on filter bank feature concatenating with pitch feature. Besides these main changes, several other modifications were also added which showed performance improvement.

The rest of this paper is organized as follows. Sections 2 and 3 introduce the acoustic modeling and the language modeling. Section 4 describes the automatic acoustic segmentation algorithm. Section 5 introduces the decoding processing which includes LM rescore and N-best ROVER procedures. Experimental results as well as discussion of the results are given in Section 6. Conclusion is given in Section 7 .

2. Acoustic Modeling

2.1. Training Corpus

Three types of data corpus were used in training the acoustic models (as shown in table 1). 81.1 hours of Wall Street Journal (WSJ), 62.9 hours of HUB4 English Broadcast news which obtained from the Linguistic Data Consortium, and 167.8 hours of processed 760 TED talks crawled from its online web site published before 2011 (with SailAlign software for extracting text-aligned acoustic segments). WSJ is read speech, HUB4 is spontaneous broadcast news speech and TED is lecture style speech.

2.2. Feature Extraction

Four types of acoustic feature sets were extracted to build acoustic models. The first type of feature set is Mel-frequency cepstral coefficient (MFCC), which was extracted with a 25 ms Hamming window that was shifted at 10 ms intervals. The MFCC feature consisted of 12 MFCCs, logarithmic power (log-power), and their first and second or-

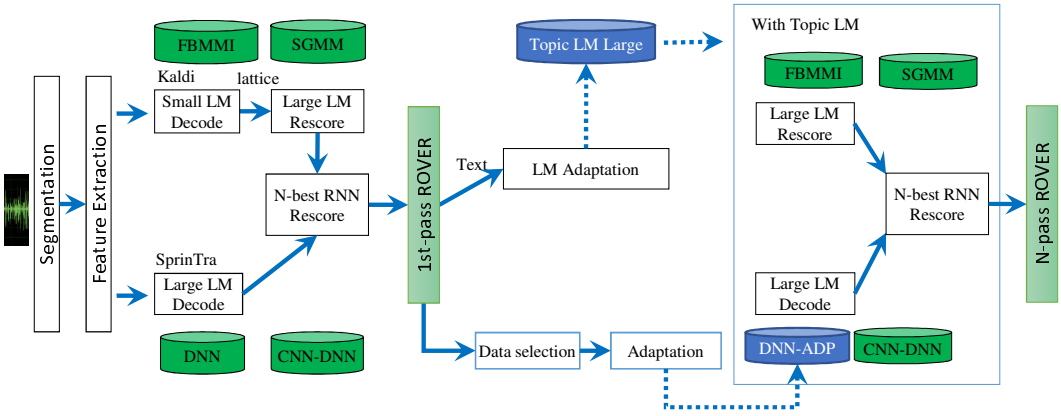

Figure 1: Overview of the NICT ASR system.

der derivatives. The dimensions of the acoustic feature vectors were 39. Then 7 adjacent frames were concatenated (3 on each side of the current frame) to make context dependent feature vectors. By applying a linear discriminate analysis (LDA), the concatenated feature vector was compressed to 40 dimensions. The 40-dimension vector was further decorrelated with a maximum likelihood linear transformation (MLLT). In addition, a feature space maximum likelihood linear regression (fMLLR) was also applied in speaker adaptive training (SAT) stage. The second type of acoustic feature set is a perceptual linear predictive cepstrum (PLP) feature, the same procedures as done on the MFCC feature were applied. The third type of feature set is log Mel filter bank feature or FBANK features. It has been shown to improve the performance of DNN based acoustic modeling than MFCC feature [4]. The fourth type of acoustic feature set is combination of FBANK feature with tonal feature. Although English is not a tonal language, it showed improvement in DNN modeling if tone feature is incorporated as an additional feature in acoustic modeling [5]. In both the third and fourth types of feature sets, the first and second time derivations of these features were utilized in acoustic modeling. In addition, because these two types of feature sets were used in different DNN architectures, their FBANK feature dimensions were also different. This will be explained in acoustic model training in next subsection.

2.3. Acoustic models and training

Four types of acoustic models were built in our system, they were FBMMI, SGMM, DNN and CNN-DNN acoustic models. To train these models, we first trained a basic context dependent triphone HMM model with GMM output probability. The final acoustic model has 7922 triphone tied states with 160180 Gaussian components. For improving the basic model, we further applied feature space maximum likelihood linear regression (fMLLR) for speaker adaptive training. This SAT-HMM/GMM model was used as a baseline for FBMMI, SGMM, DNN and CNN-DNN acoustic model training.

The FBMMI is a discriminative training with feature space boosted maximum mutual information (FBMMI) criterion [6]. The subspace GMM (SGMM) model was trained by clustering the Gaussians from the triphone HMM/GMM model. In addition, the FBMMI was also conducted on SGMM for discriminative training. The FBMMI and SGMM acoustic models were trained by two types of feature sets, MFCC and PLP. Therefore, four acoustic models were obtained.

Two types of DNN architectures were used for acoustic modeling. One is feedforward DNN (hereafter it is called as DNN). Another is convolutional DNN in which the input layer is with convolutional operator while other layers are feedforward DNN (hereafter it is called as CNN-DNN). In DNN training, a frame-based cross-entropy criterion was first applied in the first stage, then a sequential discriminative training based on a state level minimum Bayesian risk criterion (sMBR) was adopted for the second stage training [7]. In CNN-DNN training, only the frame-based cross-entropy criterion was used. For training the DNN and CNN-DNN, different types of feature sets were used. For MFCC and PLP feature sets, the DNN architecture was configured as: 300-2100*5-7922, i.e., input layer was with 300 neurons, 5 hidden layers with 2100 neurons for each, and 7922 neurons in the output layer. The input layer feature was transformed by LDA from 15 consecutive frames of either MFCC or PLP feature (from SAT-HMM/GMM model). For FBANK feature used in DNN, 24 Mel filter banks were used (hereafter as FBANK24 feature type). The DNN was configured as: 1080-2100*5-7922.

In CNN-DNN modeling, compared with DNNs, CNN restricts the network architecture with local connections and weight sharing so that it can explores local correlation in feature processing [8]. Our CNN-DNN has one convolutional layer with convolution and polling operations. The configuration of the convolutional layer as: 128 filters with filter

Table 1: Details of acoustic model training data

Corpus	Hours	Type	Data
WSJ	81.1	Read	LDC93S6B, LDC94S13B
HUB4	62.9	Broadcast	LDC97S44, LDC98S71
TED	167.8	Lecture	760 talks (Before 2011)

size and shift as 9 and 1 for each. In the pooling layer, local averaging and sub-sampling were performed to reduce the resolution of the feature map and the sensitivity of the output to input shifts and distortions. The pooling width and shift was set to 2 and 2, respectively. The output from the pooling layer was further processed with feedforward DNN with 4 hidden layers (2100 neurons in each layer), and one output layer (7922 neurons). In training the CNN-DNN, FBANK feature with tone feature set was used. 40 Mel filter banks and 3 dimensional tone features were used (hereafter as FBANK40+Pitch feature type).

2.4. Speaker Adaptation for DNN

In our system, speaker adaptation on DNN AMs were applied. The adaptation was operated on the third hidden layer of the DNNs based on our previous work [9]. The adaptation data was selected based on word confidence from decoding results (confidence threshold 0.7 was chosen in our study). Different from last year's adaptation processing, the adaptation data was selected based on the ROVER result. In order to overcome the overtraining problem in adaptation, a L2 regularization on the model parameters was utilized. 4 rounds adaptation were conducted on the DNN models. In each adaptation, the learning rate was set to 0.001, the number of training epoches were set to 20.

3. Language Modeling

3.1. Training data

Table 2 shows the data for training language models. It contains two categories of textual corpora that are allowed by the IWSLT evaluation campaign. One is in-domain corpus TED talk transcripts supplied by the IWSLT2014 committee, another are out-of-domain corpora. For the out-of-domain corpora, News Commentary V7 and Europarl V6 provided by the IWSLT2014 committee were used for LM training without selections, but English Gigawords and News Shuffle were further selected for the training. All of these data were normalized (or pre-processed) by using a non-standard-word expansion tools [10], so that all those non-standard words such as abbreviation, numbers etc, were converted to simple words. For examples, words "CO2" and "95%" were converted to "CO two" and "ninety five percent." Duplicated sentences were removed during this normalization process.

Table 2: Training data of language models.

Category	Corpus	Tokens
In-domain	TED Talks	3.2M
Out-of domain	NewsCommentary V7	4.6M
	Europarl V7	50.0M
	English Gigawords 5th ed.	2.7G
	News Shuffle	732.8M

3.2. Domain adapted n-gram LM

The first pass of speech decoding was performed using a domain adapted n-gram LM. The adapted LM was built by interpolating a in-domain n-gram and several adaptation n-grams. The in-domain n-gram was constructed by using the in-domain data, and the adaptation n-grams were constructed by using the selected sentences from the out-of-domain corpora. However, since there are many sentences in the out-of-domain that are highly mismatched to the TED domain, these sentences will be harmful to LM if they are added to training data. Therefore, we built adaptation LMs by selecting adequate training sentences from two of the out-of-domain corpora - English Gigawords and News Shuffle. Since the News Commentary data and the Europarl are relative small, no selection was conducted on them.

The sentence selection was based on a cross-entropy difference metric [11] which was biased towards sentences that were both similar to the in-domain data and unlike the average of the out-of-domain data. Here, the similarity and unlikeness were measured by the sentence entropy (or perplexity) with respect to in-domain LM and out-of-domain LM, respectively. Detailed description about this selection algorithm can be referred in [12]. Finally, about 30.0M sentences (560M tokens) from the English Gigaword data, and 7.6M sentences (143.8M tokens) were selected.

Using the SRILM toolkit [13], the modified Kneser-Ney smoothed n-grams (n=4) were constructed for in-domain LM using the TED corpus, and for adaptation LMs accordingly using the selected sentences, the News Commentary V7 data and the Europarl V7 data. The domain adapted LM was achieved by linearly interpolating these n-grams, with the development set defined in the IWSLT evaluation campaign for optimization. In all these training process, a vocabulary of 123K words from the CMU Pronunciation Dictionary [14] and the TED corpus was used.

3.3. Topic adapted n-gram LM

The second pass of speech decoding was conducted using a topic adapted LM constructed by the recognition results of the first decoding pass. The sentence selection for the topic adapted LM was conducted in the same way as for the domain adapted LM. The data sources for selection were still the English Gigawords and the News Shffled, however, the recognition results obtained from the first decoding pass were

used as the seed data for selection. The sentence cross entropy was measured between two n-gram LMs, one was built by using recognition results of all talks in the first decoding pass, another was built by using 2000 sentences randomly selected from the resource data. Finally, 61.7M sentences (246.7M tokens) were selected from the English Gigawords, and 3.8M sentences (65.7M tokens were selected from the News Shuffle. Two n-grams (n=4) were built by using these sentences individually. The topic adapted LM was then constructed by linearly interpolating these two LMs, other two LMs built respectively by the News Commentary and Europarl, (for these two corpora, no sentence selections are conducted with them), and the in-domain LM.

3.4. RNNLM

In this system, N-best list rescoring was adopted and performed using a recurrent neural network(RNN) LM [15]. In our RNN, the number of units in the hidden layer and classes in the output layer were 480 and 300, respectively. Back-propagation Through Time (BPTT) with truncated time order 5 was used in RNN training. The training data for the RNN was the same as that for the domain adapted n-gram LM described above. To decrease the training time, only one-tenth of the selected out-of-domain sentences were used for the training.

4. Automatic Segmentation

In this year's evaluation, the whole TED talks in the test data set were provided without any acoustic segmentation information. For convenience of decoding and rescoring, acoustic segmentation was first done. A combination method of a voice activity detection (VAD) algorithm and acoustic event detection (AED) algorithm was utilized for this purpose. In designing the VAD algorithm, signal power energy and spectral centroid features were used. In AED, five GMMs corresponding to five acoustic events (speech, music, applause, laugh and background noise) mostly appeared in lecture speech were trained in this study. MFCC feature was used in GMM training, and the diagonal GMM consists 16 mixtures was used in AED. The acoustic segmentation was done based on merging the detection results of VAD and GMM. In merging, a hang-over scheme with minimum durations of non-speech event as 800ms, and minimum duration of speech event as 160ms was applied. Based on the segmented utterances, the feature extraction, decoding and ROVER were carried out in recognition experiments.

5. Decoding and ROVER

5.1. Decoding System

Two types of WFST-based decoders were used. One is Kaldi decoder, the other is NICT SprinTra decoder. The Kaldi decoder was used for FBMMI and SGMM acoustic model based decoding, and SprinTra decoder was utilized for DNN and CNN-DNN acoustic model based decoding.

In decoding with Kaldi decoder, a small 4-gram LM was first used to produce word lattice. Then a large 4-gram LM was applied for rescoring on the word lattice. For improving the performance, the RNN LM was further applied on the N-best list for rescoring. In decoding with NICT SprinTra decoder, the large 4-gram LM was directly used. Based on the decoding word lattice, RNN LM was also used on the N-best list for rescoring.

5.2. N-best ROVER

A N-best recognizer output voting error reduction (ROVER) algorithm was applied to combine all the subsystems for further improving the performance. This year, subsystems with four types of acoustic models (FBMMI, SGMM, DNN and CNN-DNN) and four types of feature sets (MFCC, PLP, FBANK24, FBANK40+Pitch) were combined in ROVER processing. For each subsystem, 50-best lists from 4-gram LM and RNN LM rescoring processing were used. In ROVER, the combination weights were selected based on our experimental results on the development data set.

6. Experimental Results

6.1. DNN Speaker Adaptation

The algorithm of speaker adaptation used in this year is similar to last year's system. But the adaptation data selection is different from last year's system. In last year's system, the adaptation data set was picked up based on the DNN decoding result. Considering that ROVER result is always better than one of the DNN decoding result, the adaptation data was selected based on the ROVER result in this year. For comparison of the two adaptation data selection methods, we showed the results in Figure 2. The decoding/rescoring results are also included for comparison. In our 2013 system, after the first pass ROVER, topic adaptation on LM was conducted. With the adapted LM, we could obtain 0.4% improvement for both 2011 and 2012 test data sets on DNN based decoding. Then the adaptation data was selected based on this DNN decoding result. In this year, we simply changed the adaptation data selection method based on word confidence calculated in the ROVER step. From the decoding results, 0.7% and 0.9% improvements were obtained for 2011 and 2012 test data sets, respectively. With this new process, our speaker adaptation on DNN can be done for multiple rounds for obtaining better performance. Table 3 shows the results of N-rounds DNN speaker adaptation process. The results showed that consistent improvements were obtained with 4-rounds DNN adaptation for each feature set separately. However, no further improvement was obtained for ROVER result with fifth round adaptation.

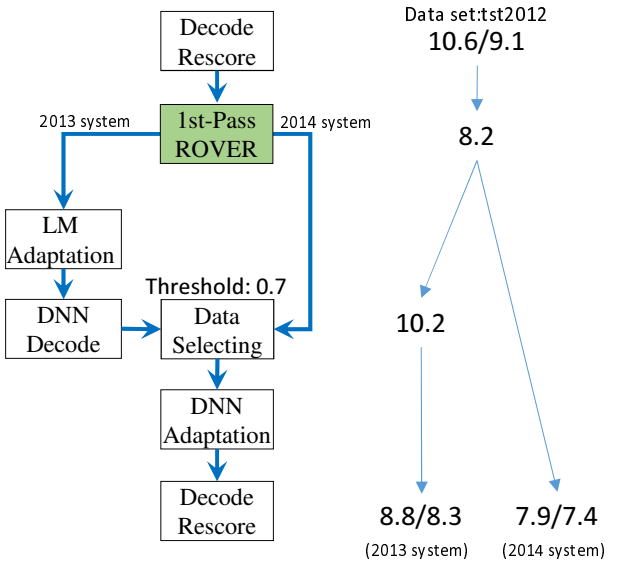

Figure 2: The adaptation process and evaluation results (WER %) of 2013, 2014 system. The feature of decoding/rescoring results are MFCC, The ROVER consists of FBMMI, SGMM, DNN acoustic models with feature MFCC and PLP.

Table 3: Contribution of N-rounds adaptation; Decoding/rescoring results are listed for DNN with feature MFCC, PLP and FBANK; The ROVER consits of all the subsystems.

subsystem	MFCC	PLP	FBANK	ROVER
DNN-baseline	16.0/14.8	16.3/15.3	15.2/14.6	12.7
1st round	12.3/11.8	12.3/11.9	12.7/12.2	11.6
2nd round	11.7/11.4	11.7/11.4	11.9/11.6	11.2
3rd round	11.5/11.3	11.4/11.2	11.6/11.4	11.1
4th round	11.3/11.2	11.3/11.1	11.4/11.3	11.1

6.2. Searching Beam and ROVER Weights

Increasing the searching beam width in decoding always helps to improve the performance but at the cost of increasing searching time. In our experiments, we set beam width to 13 (the same as in last year) for both Kaldi and SprinTra decoders in the first few steps of decoding. In the last DNN adaptation step, the beam width was set to 17 which resulted in 0.1% improvement in the WER.

In ROVER processing, the combination weights were set as 1:1:2 for FBMMI, SGMM and DNN in last year. After adding the DNN-FBANK24 and CNN-DNN acoustic model based subsystems, the combination weights were re-investigated based on the development data set. Different combination weight sets were set for ROVER: 1:1:3:3 for FBMMI, SGMM, DNN, CNN-DNN for the first pass ROVER and 1:1:7:1 for N-round pass ROVER (N=2,3,4,5).

6.3. Contributions of Subsystems

Table 4 shows the results on 2013 test data set with different combinations of subsystems in the first pass ROVER. With

Table 4: Contribution of each subsystems(first pass ROVER); data set: 2013 test data set

subsystem	sys1	sys2	sys3	sys4	sys5	sys6	sys7	sys8
FBMMI	○		○		○	○		○
SGMM	○		○		○	○		○
DNN-mfcc		○	○	○	○	○	○	○
DNN-plp		○	○	○	○	○	○	○
DNN-fbank				○		○	○	○
CNN-DNN					○		○	○
WER(%)	18.1	14.5	13.8	13.4	13.1	13.1	12.9	12.7

Table 5: Contribution of each subsystems(the fifth pass ROVER), with topic adapted LM and speaker adaptation for DNN models; data set: 2013 test data set

subsystem	sys1	sys2	sys3	sys4	sys5	sys6	sys7	sys8
FBMMI	○		○		○	○		○
SGMM	○		○		○	○		○
DNN-mfcc		○	○	○	○	○	○	○
DNN-plp		○	○	○	○	○	○	○
DNN-fbank				○		○	○	○
CNN-DNN					○		○	○
WER(%)	17.8	11.1	11.1	11.0	11.1	11.1	11.1	11.0

the subsystems used in last year (sys3), we obtained 13.8% WER. 1.1% absolute improvement was obtained after adding the DNN-FBANK24 and CNN-DNN based subsystems in ROVER. Also from this table, we can see that although DNN and CNN-DNN subsystems obtained quite low WER, taking the FBMMI and SGMM based subsystems in ROVER processing still helped to improve the performance (about 0.2% improvement).

Table 5 shows the results of the fifth pass ROVER step. In this step, the LM and DNN acoustic model were adapted with the methods described in the previous section. Different to the first pass ROVER result, we obtained almost the same result by only combing DNN acoustic model based subsystems with or without the FBMMI, SGMM and CNN-DNN based subsystems.

6.4. Summary of Results

Table 6 shows the summary of our ASR system comparing with last year's official best rest for 2011, 2012, and 2013 test data sets. Compared to last year's official result, this year ASR approach achieved a better performance. The automatic segmentation, combination of new subsystems in ROVER, and multi-rounds speaker adaptation contributed the most of the improvements. After 4-rounds speaker adaptation on DNN acoustic models, there was no further improvement in final ROVER processing. For this year's test set, we obtained 8.4% WER.

Table 6: The final results (WER %) of the test sets: 2011, 2012, 2013 and 2014.(* means using NICT's references)

	tst2011	tst2012	tst2013	tst2014
Official best(2013)	7.9	8.6	13.5	-
NICT 2014	6.5*	7.0*	10.6	8.4

7. Conclusions

In this study, we describe our ASR system for the IWSLT 2014 evaluation campaign. Our ASR system consists of four types of acoustic models (FBMMI, SGMM, DNN and CNN-DNN), four types of acoustic features (MFCC, PLP, FBANK24 and FBANK40+Pitch), and two types of LMs (N-gram and RNN). Several improvements were conducted, such as new acoustic models, automatic segmentation, and DNN speaker adaptation. The results of our proposed approaches demonstrate a better performance than that of last year.

8. References

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

[2] H. Yamamoto, Y. Wu, C. Huang, X. Lu, P. R. Dixon, S. Matsuda, C. Hori, and H. Kashioka, "The NICT ASR System for IWSLT2012," In *Proc. of IWSLT*, 2012.

[3] C. Huang, P. R. Dixon, S. Matsuda, Y. Wu, X. Lu, M. Saiko, and C. Hori, "The NICT ASR system for IWSLT 2013," In *Proc. of IWSLT*, 2013.

[4] L. Deng and J. Li and J-T. Huang, K. Yao, D. Yu, F. Seide,M. Seltzer, G. Zweig, X. He, J. Williams, Y. Gong, and A. Acero, "Recent advances in deep learning for speech research at Microsoft," In *Proc. of ICASSP*, 2013.

[5] X. Lei, M-Y Hwang, and M. Ostendorf, "Incorporating tone-related MLP posteriors in the feature representation for Mandarin ASR," In *Proc. Eur. Conf. Speech Communication Technology*, 2005.

[6] D. Povey, S. M. Chu, J. Pelecanos, and H. Soltau, "Approaches to Speech Recognition based on Speaker Recognition Techniques," Chapter in forthcoming GALE book.

[7] K. Vesely, A. Ghoshal, L. Burget, and D. Povey, "Sequence-discriminative training of deep neural networks," in *Proc. of Interspeech*, 2013.

[8] X. Hu, X. Lu and, and C. Hori, "Mandarin Speech Recognition Using Convolution Neural Network With Augmented Tone Features," In *Proc. of ISCSLP*, 2014.

[9] T. Ochiai, S. Matsuda, X. Lu, C. Hori, and S. Katagiri, "Speaker adaptive training using deep neural networks," in *Proc of ICASSP*, Italy, 2014.

[10] R. Sproat, A. W. Black, S. Chen, S. Kumar, M. Ostendorf, and C. Richards, "Normalization of Non-Standard Words," in Computer Speech and Language , pp.287-333, 2001.

[11] R. C. Moore, and W. Lewis, "Intelligent Selection of language Model Training Data," in *Proc. of ALC*, 2010.

[12] P. Bell, H. Yamamoto, P. Swietojanski, Y. Z. Wu, F. McInnes, C. Hori, and S. Renals, "A Lecture Transcription System Combining Neural Network Acoustic and Language Model," in *Proc. of Interspeech*, 2013.

[13] A. Stolcke, "SRILM - An extensible Language Modeling Toolkit," in *Proc. of ICSLP*, 2002.

[14] http://www.speech.cs.cmu.edu/cgi-bin/cmudic

[15] T. Mikolov, M. Cettolo, L. Burget, J. Cernnokcy, and S. Khudanpur, "Recurrent Neural Network Based Language Model," in *Proc. of Interspeech*, 2010.