The Speech Recognition Systems of IOIT for IWSLT 2014

Quoc Bao Nguyen[1], *Tat Thang Vu*[2], *Chi Mai Luong*[2]

[1] University of Information and Communication Technology, Thai Nguyen University, Vietnam
[2] Institute of Information and Technology (IOIT), Vietnamese Academy of Science and Technology (VAST)
nqbao@ictu.edu.vn, {vtthang,lcmai}@ioit.ac.vn

Abstract

This paper describes the speech recognition systems of IOIT for IWSLT 2014 TED ASR track. This year, we focus on improving acoustic model for the systems using two main approaches of deep neural network which are hybrid and bottleneck feature systems. These two subsystems are combined using lattice Minimum Bayes-Risk decoding. On the 2013 evaluations set, which serves as a progress test set, we were able to reduce the word error rate of our transcription systems from 27.2% to 24.0%, a relative reduction of 11.7%.

1. Introduction

The International Workshop on Spoken Language Translation (IWSLT) is a yearly scientific workshop, associated with an open evaluation campaign on spoken language translation. One part of the campaign focuses on the translation of TED Talks, which are a collection of public lectures on a variety of topics, ranging from Technology, Entertainment to Design. As in the previous years, the evaluation offers specific tracks for all the core technologies involved in spoken language translation, namely automatic speech recognition (ASR), machine translation (MT), and spoken language translation (SLT).

The goal of the ASR track is the transcription of audio coming from unsegmented TED and TEDx talks, in order to interface with the machine translation components in the speech-translation track. The quality of the resulting transcriptions are measured in word error rate (WER).

In this paper we describe the speech recognition systems which we participated in the TED ASR track of the 2014 IWSLT evaluation campaign. This year, our system is a further development of our last year's evaluation system [1], and focuses on improving acoustic model using deep neural network. There are two main approaches for incorporating artificial neural networks in acoustic modeling today: hybrid systems and tandem systems. In the hybrid approach, a neural network is trained to estimate the emission probabilities for Hidden Markov Models (HMM) [2]. In contrast, tandem systems use neural networks to generate discriminative features as input values for the common combination of Gaussian Mixture Models (GMM) and HMMs. One of the common tandem system uses the activations of a small hidden layer ("bottleneck features", BNF [3]).

The organization of the paper is as follows. Section 2 describes the data that our system was trained on. This is followed by Section 3 which provides a description of the way to extract deep bottleneck features. An overview of the techniques used to build our acoustic models is given in Section 4. Dictionary and language model are presented in Section 5. We describe the automatic segmentation process in Section 6. Our decoding procedure and results are presented in Section 7.

2. Training Corpus

For acoustic model training, we used TED talk lectures (http://www.ted.com/talks) as training data. Approximately 220 hours of audio, distributed among 920 talks, were crawled with their subtitles, which were deliberately used for making transcripts. However, the provided subtitles do not contain the correct time stamps corresponding with each phrase as well as the exact pronunciation for the words spoken, which lead to the necessity for long-speech alignment.

Segmenting the TED data into sentence-like units used for building a training set was performed with the help of SailAlign tool [4] which helps us to not only acquire the transcript with exact timing, but also to filter non-spoken sounds such as music or applauses. A part of these noises are kept for noise training while most of them are abolished. After that, the remained audio used for training consists of around 160 hours of speech.

3. Deep Bottleneck Features

In this work, we applied the deep neural network architecture for bottleneck feature extraction (DBNFs) as in [5] [6] and depicted in Figure 1. The network consists of a variable number of moderately large, fully connected hidden layers and a small bottleneck layer which is followed by an additional hidden layer and the final classification layer.

The Mel-frequency cepstral coefficients (MFCCs) features were used as input of deep neural network, which contain 39 coefficients including 12 cepstral coefficients, 1 energy coefficient added with delta and double-delta features were extracted after windowing with the window size of 25 milliseconds and frame shift of 10 milliseconds. Then they were pre-processed using the approach in [7] that is called

splicing speaker-adapted features with 40 dimensions. This features for each frame were stacked with 9 adjacent samples, resulting in a total of 360 dimensions. For pre-training the stack of auto-encoders, mini-batch gradient descent with a batch size of 128 and a learning rate of 0.01 was used. Input vectors were corrupted by applying masking noise to set a random 20% of their elements to zero. Each auto-encoder contained 1024 hidden units and received 1 million updates before its weights were fixed and the next one was trained on top of it.

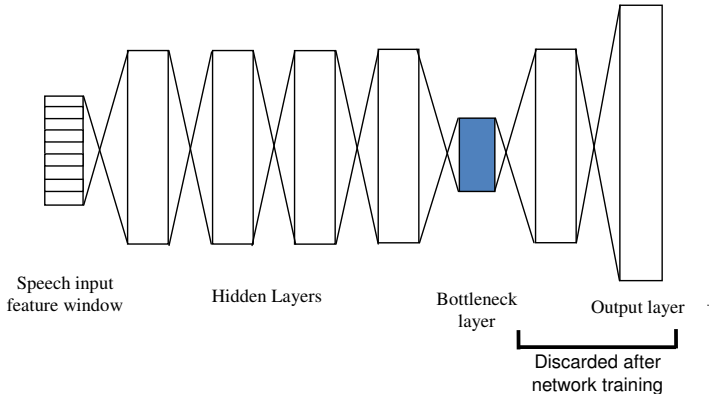

Figure 1: Deep Network Architecture for Bottleneck Features

The remaining layers were then added to the network, with the bottleneck layer consisting of 39 units, another hidden layer and output layer containing 4,500 context-dependent HMM states. Again, gradients were computed by averaging across a mini-batch of training examples; for fine-tuning, we used a larger batch size of 256. The learning rate was decided by the newbob schedule: for the first epoch, we used 0.008 as the learning rate, and this was kept fixed as long as the increment in cross-validation frame accuracy in a single epoch was higher than 0.5%. For the subsequent epochs, the learning rate was halved; this was repeated until the increase in cross-validation accuracy per epoch is less than a stopping threshold, of 0.1%. The activations of the 39 bottleneck units are stacked over an 9-frame context window and reduced to a dimensionality of 40 using linear discriminant analysis (LDA) and maximum likelihood linear transformation (MLLT).

4. Acoustic Model

4.1. Baseline Acoustic Model

Baseline HMM/GMM acoustic model were performed with the Kaldi developed at Johns Hopkins University [8]. Nine consecutive MFCC feature frames were spliced to 40 dimensions using linear discriminant analysis (LDA) and maximum likelihood linear transformation (MLLT) that is a feature orthogonalizing transform, was applied to make the features more accurately modeled by diagonal-covariance Gaussians.

All models used 4,500 context-dependent state and 96,000 Gaussian mixture components. The baseline systems were built, follow a typical maximum likelihood acoustic training recipe, beginning with a flat-start initialization of context-independent phonetic HMMs, followed by tri-phone system with 13-dimensional MFCCs plus its deltas and double-deltas and ending with tri-phone system and LDA+MLLT.

4.2. Hybrid Acoustic Model

For the hybrid network training, we used the same techniques that described in the deep bottleneck feature section. The network architecture, we settled with a stacked 5 auto-encoders containing 1024 units each. Its input was used the same with DBNFs network, MFCCs feature were pre-processed and stacked over a 9 adjacent frames. 4,500 context-dependent target states were used for supervised training that is the number of tied states in the respective baseline systems.

5. Dictionary and Language Model

The word set contains 131,137 words. The lexicon was built based on the Carnegie Mellon University (CMU) Pronouncing Dictionary v0.7a; the phoneme set contains 39 phonemes. This phoneme (or more accurately, phone) set is based on the ARPAbet symbol set developed for speech recognition uses. The vowels may carry lexical stress, ranging from no stress, primary stress to secondary stress.

For language modeling, the in-domain data was provided by organizer and 1/8 of Giga corpus was also utilized by filtering it according to the Moore-Lewis approach [9]. Both two datasets were normalized using the normalization toolkit from CMU. The vocabulary used to train language models is the same as in the lexicon. The training data for language model is summarized in Table 1.

Table 1: *Training data for language modeling for English ASR Task.*

Data	Number of sentences	Number of tokens
TED	156,460	2,708,816
1/8 Giga	2,565,687	56,488,064

We trained 3-gram language model using SRILM toolkit with the modified interpolated Knesey-Ney smoothing technique [10] from each of data set. These were then combined using linear interpolation as follows:

$$P(w|h) = \lambda_1 P_1(w|h) + \lambda_2 P_2(w|h) + ... + \lambda_n P_n(w|h)$$

Where $\lambda_1, \lambda_2, ..., \lambda_n$ are the interpolation weights which were chosen to maximize the likelihood of a held out TED data set.

6. Auto Segmentation

The evaluation data has only provided unsegmented audio data since last year. Therefore, in our works the LIUM Diarization toolkit [11] was used to divide the talk into small sentence-like segments. Figure 2 provides a general description on the diarization process. First, 13 MFCC features were extracted from the long audio file. After that, a Viterbi decoding is performed to generate a segmentation. Some of segment boundaries produced by the Viterbi decoding fall within words. These boundaries are adjusted by applying a set of rules defined experimentally. Detection of gender and bandwidth is then done using a GMM for each of the 4 combinations of gender (male / female) and bandwidth (narrow / wide band). Finally, GMMbased speaker clustering is carried out to map each speech segment to the corresponding speaker.

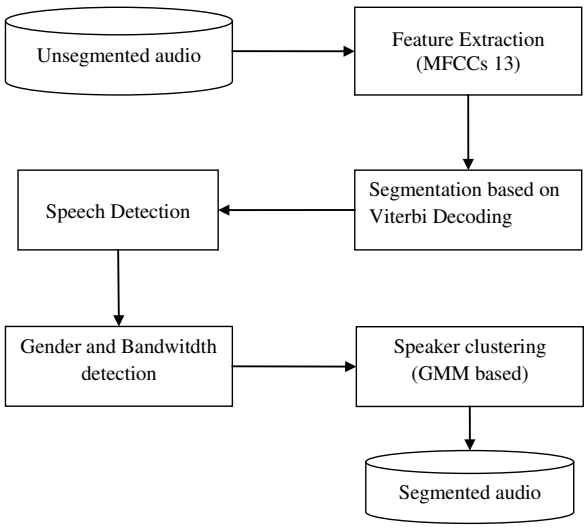

Figure 2: Deep Network Architecture for Bottleneck Features

Comparing automatic and manual segmentation, the disparity in word error rates is disclosed in Table 2. It is notable that the automatic speech detection caused approximately 2 percent loss of the spoken audio, resulted in inevitably decreasing the error rates, presented by deletions. Experiments conducted with tst2013 data illustrated that the WER increased 10% relatively, compared with the same data sets which are manually segmented. The segmentation cannot be guaranteed to be precise at the beginning or the end of the sentence, the output segments are sometime incomplete sentence, or incomplete phrases, which affects recognition results. Last year, we proposed a type of recurrent neural network language model(RNNLM) [1] to resolve this problem. We did not use RNNLM this year because of time consuming.

7. Decoding Procedure and Results

During development, we evaluated our system using the 2012 development set and 2013 test set that released by the IWSLT organizers.

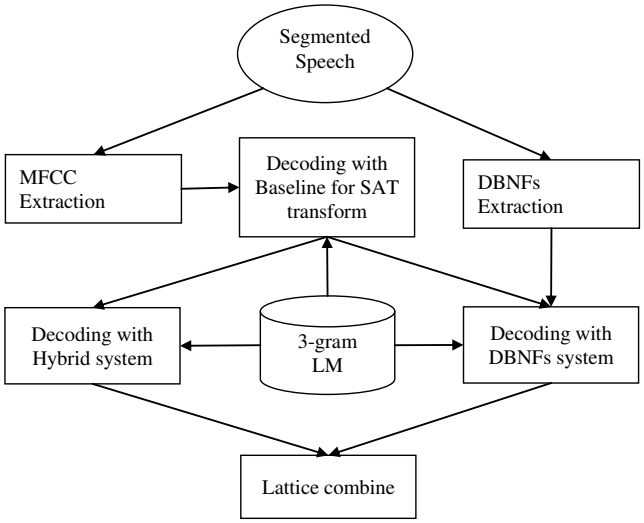

Figure 3: The full decoder architecture

Figure 3 shows the complete decoding architecture. After feature extraction step, followed by decoding with the baseline system to estimate the transformations for speaker adaptation (fMLLR algorithm), we operate two parallel decoding sequences for the tandem and hybrid acoustic models. For each model, the complete process consists of a decoding with the trigram LM using Kaldi decoder tool. Lattices output from the this pass were combined using Lattice Minimum Bayes-Risk (MBR) decoding as described in [12]

Table 2: *English ASR results for various acoustic models and segmentation types (manual, auto)*

System	**WER(%)**		
	dev2012	**tst2013**	**tst2013 auto**
Baseline	30.0	36.1	–
Last year	22.9	29.5	27.2
DBNFs(S1)	19.5	23.8	25.7
Hybrid(S2)	20.0	23.6	25.3
S1+S2	18.7	22.7	24.0

Table 2 lists the performance of our systems in terms of the word error rate (WER). Regarding the performance of the baseline system, the WER is 30.0% on dev2012 and 36.1% on tst2013. The second row is the number of the best system from last year [1] where we applied state-of-the-art techniques for acoustic model without deep neural network. Results for applying deep bottleneck features are listed on third

row of the table. As we can see the results, the DBNF numbers are about 10% absolute (about 33% relative) better than the baseline numbers on both sets. The hybrid DNN/HMM combination also outperforms baseline setup with similar results to DBNFs number. The last row on the table shows the final system combination results of DBNFs and Hybrid systems that gives a further 1% absolute WER reduction as compared to the best single system.

8. Conclusions

In this paper, we presented our English LVCSR systems, with which we participated in the 2014 IWSLT evaluation. The acoustic model was improved using deep neural network for this year evaluation. On the 2012 development set for the IWSLT lecture task our system achieves a WER of 18.7%, and a WER of 24.0% on the 2013 test set.

In the future, we intend to improve language model using deep neural network as in [1] as well as apply a hybrid DNN on top of deep bottleneck features [6] and multi-lingual network training approaches [13] to improve acoustic model for the systems.

9. Acknowledgements

This work was partially supported by Project: "Development of spoken electronics newspaper system based on Vietnamese text-to-speech and web-based technology", VAST01.02/14-15

10. References

[1] N. Q. Pham, H. S. Le, T. T. Vu, , and C. M. Luong, "The speech recognition and machine translation system of ioit for iwslt 2013," in *Proceedings of the International Workshop for Spoken Language Translation (IWSLT)*, 2013.

[2] H. A. Bourlard and N. Morgan, *Connectionist Speech Recognition: A Hybrid Approach*. Norwell, MA, USA: Kluwer Academic Publishers, 1993.

[3] F. Grezl, M. Karafiat, S. Kontair, and J. Cernocky, "Probabilistic and bottle-neck features for lvcsr of meetings," in *Acoustics, Speech and Signal Processing (ICASSP), 2007 IEEE International Conference on. IEEE*, 2007, pp. V–757 – IV–760.

[4] A. Katsamanis, M. Black, P. G. Georgiou, L. Goldstein, and S. S. Narayanan, "Sailalign: Robust long speech-text alignment," in *Proc. of Workshop on New Tools and Methods for Very-Large Scale Phonetics Research*, jan 2011.

[5] J. Gehring, Y. Miao, F. Metze, and A. Waibel, "Extracting deep bottleneck features using stacked auto-encoders," in *ICASSP2013*, Vancouver, CA, 2013, pp. 3377–3381.

[6] Q. B. Nguyen, J. Gehring, K. Kilgour, and A. Waibel, "Optimizing deep bottleneck feature extraction," in *Computing and Communication Technologies, Research, Innovation, and Vision for the Future (RIVF), 2013 IEEE RIVF International Conference on*, Nov 2013, pp. 152–156.

[7] S. P. Rath, D. Povey, K. Vesely, and J. Cernocky, "Improved feature processing for deep neural networks." in *INTERSPEECH*. ISCA, 2013, pp. 109–113.

[8] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, "The kaldi speech recognition toolkit," in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. IEEE Signal Processing Society, Dec. 2011, iEEE Catalog No.: CFP11SRW-USB.

[9] R. C. Moore and W. Lewis, "Intelligent selection of language model training data," in *Proceedings of the ACL 2010 Conference Short Papers*, ser. ACLShort '10. Stroudsburg, PA, USA: Association for Computational Linguistics, 2010, pp. 220–224.

[10] S. F. Chen and J. Goodman, "An empirical study of smoothing techniques for language modeling," in *Proceedings of the 34th Annual Meeting on Association for Computational Linguistics*, ser. ACL '96. Stroudsburg, PA, USA: Association for Computational Linguistics, 1996, pp. 310–318.

[11] S. Meignier and T. Merlin, "Lium spkdiarization: an open source toolkit for diarization," in *in CMU SPUD Workshop*, 2010.

[12] H. Xu, D. Povey, L. Mangu, and J. Zhu, "Minimum bayes risk decoding and system combination based on a recursion for edit distance," *Computer Speech & Language*, vol. 25, no. 4, pp. 802–828, 2011.

[13] Q. B. Nguyen, J. Gehring, M. Muller, S. Stuker, and A. Waibel, "Multilingual shifting deep bottleneck features for low-resource asr," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, May 2014, pp. 5607–5611.