INTERSPEECH 2016
September 8–12, 2016, San Francisco, USA

Investigation of Semi-supervised Acoustic Model Training based on the Committee of Heterogeneous Neural Networks

Naoyuki Kanda, Shoji Harada, Xugang Lu, Hisashi Kawai

National Institute of Information and Communications Technology, Japan
naoyuki.kanda,show.harada,xugang.lu,hisashi.kawai}@nict.go.jp

Abstract

This paper investigates the semi-supervised training for deep neural network-based acoustic models (AM). In the conventional self-learning approach, a "seed-AM" is first trained by using a small transcribed data set. Then, a large untranscribed data set is decoded by using the seed-AM to create a transcription, which is finally used to train a new AM on the entire data. Our investigation in this paper focuses on the different approach that uses additional complementary AMs to form a committee of label creation for untranscribed data. Especially, we investigate the case of using heterogeneous neural networks as complementary AMs, and the case of intentional exclusion of the primary seed-AM from the committee, both of which could enhance the chance to find more informative training samples for the seed-AM. We investigated those approaches based on Japanese lecture recognition experiments with 50-hours of transcribed data and 190-hours of untranscribed data. In our experiment, the committee-based approach showed significant improvements in the word error rate, and the best method finally recovered 75.2% of the oracle improvement with full manual transcription, while the conventional self-learning approach recovered only 32.7% of the oracle gain.

Index Terms: Semi-supervised training, deep neural network, acoustic model

1. Introduction

In almost all cases, the larger the training data are, the better the acoustic models (AMs) will be. However, manual transcription of speech corpus is expensive and time-consuming. Thus, semi-supervised learning [1] of AMs, in which AMs are trained on small transcribed data and large untranscribed data, has been gathering many attentions.

One widely-used approach of semi-supervised learning is self-learning [2, 3, 4, 5, 6, 7]. In the self-learning approach, a seed-AM is first trained by using a small transcribed data. Then, a large untranscribed data is decoded by using the seed-AM to create automatic transcription. Finally, the transcribed data and untranscribed data with automatic transcription is used in combination to train a new AM. Because automatic transcription contains many errors, some form of confidence measure (CM) is often used in combination to select reliable parts of the transcription. One problem with self-learning with CM-based data selection is, however, that "the data with high-CM" is often "the data that the seed-AM has already been well-trained for". Such data has little information for updating the seed-AM.

In this paper, we investigate a different approach, in which additional complementary AMs are introduced besides the primary AM to form a committee of label creation for untranscribed data. By using complementary AMs, the possibility to produce training samples that are not covered by the seed-AM is enhanced. Note that, for Gaussian mixture model (GMM)-based AMs, this kind of semi-supervised training has been explored at various levels of the system combination, such as the systems with different feature types [8], different training data sets [9], etc. However, in terms of deep neural network (DNN)-based AMs [10, 11], most studies on semi-supervised training were still based on the self-learning approach [4, 5, 6, 7, 12, 13] and only limited studies which investigated the combination with GMM-based AMs [14, 15, 16, 17] were reported.

Different from the previous reports, we investigate the use of heterogeneous neural networks as complementary AMs. As a notable progress of AMs, various extensions of network architecture were recently studied, such as convolutional neural networks (CNN) [18], sigmoid-unit-type recurrent neural networks (SigRNN) [19, 20, 21, 22, 22] and long short-term memory (LSTM) neural networks [23, 24, 25, 26, 27]. While the main purposes of these studies were to improve the accuracies by using single models, these heterogeneous models could bring very large diversity to produce training samples that the primary AM could not produce by itself. We also investigate a more radical approach, which intentionally excludes the primary AM from the committee of label creation. Since "the data that are not covered by the primary AM" are basically (except by accident) mis-recognized in the transcription by the primary AM, simply excluding such transcription could enhance the chance to produce informative training samples from the viewpoint of the primary AM. This approach is known as "cross-adaptation" [28, 29] and "cross-validation adaptation" [30] in the speaker adaptation field, and we investigate the effect of the approach for semi-supervised training settings. We experimentally show that these approaches make the semi-supervised training of DNN-AM significantly (more than double) effective compared to the self-learning or the naive combination of homogeneous networks.

2. Semi-supervised training

In this study, we compare two approaches of semi-supervised training. One is the conventional self-learning approach, and one is the committee-based approach.

2.1. Self-learning

The training protocol of self-learning is as follows.

1. A seed AM is trained by using a small supervised data set.
2. A large unsupervised data set is decoded by using the seed AM to create an automatic transcription.
3. Frame-level CM in the automatic transcription is calcu-

http://dx.doi.org/10.21437/Interspeech.2016-72

lated, and data frames that are above threshold in CM are selected as training samples. In this paper, we used posterior-based CM after minimum Bayes risk decoding [4].

4. Finally, the supervised data set and selected portion of the unsupervised data set are jointly used for re-training of the seed AM.

2.2. Committee-based semi-supervised training

The training protocol of the committee-based semi-supervised training that we focus on in this paper is as follows.

1. A primary AM and additional complementary AMs are trained by using a small supervised data set.
2. A large unsupervised data is decoded by using each AM to create multiple automatic transcriptions.
3. Multiple transcriptions are then lined up, and data frames that the labels from multiple results agree on a certain level are selected as training samples. In this study, we used the frame-level coincidence of hidden Markov model (HMM) state in each transcription. Namely, if the HMM state of time frame t in all or some transcriptions are matched, the data in time t is used as the training data with the label of the corresponding HMM state [1].
4. Finally, the supervised data set and selected portion of the unsupervised data set are jointly used for re-training of the primary AM.

To make the experiment simple, we mixed-up the supervised and unsupervised training samples in random order, without applying any importance weighting for each sample. Various combinations of models/agreements options were tested, which will be discussed in the experiment section.

3. Neural Network based-AM

In this paper, we used a hybrid framework of neural networks and HMMs (DNN-HMM hybrid framework) [10, 11]. We used three types of neural network-based AMs; DNN, SigRNN and LSTM.

DNN is an artificial neural network with multiple hidden layers. When applying DNNs for acoustic modeling, each node in the output layer is set to correspond to a HMM state. Because DNNs originally represent the posterior probability of HMM-state given observation, the emission probability for decoding is estimated by applying a Bayes conversion in the DNN-HMM hybrid framework [10, 11].

SigRNN, also known as Elman network [31], is an artificial neural network with additional recurrent connection in a hidden layer, which enables the model to represent long-term time dependencies. Recently, some researchers proposed deep recurrent neural networks [20, 21, 22], consisting of multiple recurrent layers, sometimes with additional non-recurrent layers. In this paper, we use deep recurrent neural networks in which each hidden layer has a recurrent connection.

[1] We assume each transcription has an alignment information of the HMM state. We also assume the label for AM training corresponds to a HMM state.

LSTM [32] enhances RNN by incorporating cells to preserve long-term memory. In LSTM, three gating units – input, output and forget gates – control the information flow in the cells. In most of recent cases [23, 24, 25, 26, 27], multiple LSTM layers were stacked to obtain better results. In this paper, we also use the deep LSTM, which consists of multiple LSTM layers.

4. Experiment

4.1. Evaluation settings

Evaluations were conducted using lecture recordings from the "Corpus of Spontaneous Japanese (CSJ) [33]", which is one of the most famous evaluation set of Japanese speech recognition [34, 35]. The corpus contains three types of data, each comprising 10 lecture recordings. The durations of evaluation set 1, set 2 and set 3 are 2.0 hours, 2.1 hours and 1.5 hours, respectively. Besides the three evaluation sets, we picked up 10 lecture recordings (2.1 hours) as the development set to tune system parameters.

As supervised (=transcribed) training data for the acoustic models, we randomly picked up 50-hours (200 talks) of "academic lecture" recordings of the CSJ. In addition, remaining 190-hours (757 talks) of academic lecture recordings of CSJ were used as unsupervised (=untranscribed) data.

As a primary acoustic model of our investigation, a DNN acoustic model with 5 hidden layers, each comprising 1,024 nodes, was trained using 50-hours of supervised data. The output layer had 4,086 nodes, which corresponded to context-dependent phone HMM states. As acoustic features, 72 dim filter-bank features (24 filter-bank features, delta coefficients and delta-delta coefficients) with mean and variance normalization per speaker were used. We concatenated the features of both the previous and following seven frames (15 frames of features in total) when inputting to DNNs. The DNN was initialized using the discriminative pre-training method [36] and was fine-tuned using the standard stochastic gradient descent (SGD) based on the cross-entropy loss (CE) criterion.

In addition to the primary DNN, we trained two complementary AMs with the same architecture (5 hidden layers, 1,024 nodes), but with different input features. One is a DNN model with 39 dim MFCC features (13 MFCC features, delta coefficients and delta-delta coefficients). The other is a DNN model with 39 dim PLP features (13 PLP features, delta coefficients and delta-delta coefficients). Both models were trained by using the same recipe that was used for the baseline model.

Furthermore, we trained two complementary AMs with different network architectures. The first one is a SigRNN model with 5 hidden layers, each of which had 512 nodes. The second one is a deep LSTM model with 2 LSTM layers, each comprising 512 nodes. Both models were trained based on the same recipe as follows. The output layer had 4,086 nodes, as in the DNN models. The 72 dim filter-bank features were used as acoustic features. When features were fed into the SigRNN or LSTM, the prior and subsequent three frames of features were concatenated. Additionally, we delayed the reference label by four frames. Consequently, these models could observe seven future frames to predict the reference label. We used the sparse initialization technique [37] with no pre-training. Both models were trained by one-sample backpropagation through time (BPTT) with pseudo shuffling technique [22] with a nine-frame back step.

For the language model (LM) for decoding, we trained a

Table 1: *WER of various neural network based AMs trained by 50-hours of supervised data.*

Model	WER (%)			
	E1	E2	E3	E (ave)
Fbank DNN	17.44	**15.04**	17.04	16.51
MFCC DNN	18.05	15.54	17.07	16.89
PLP DNN	18.11	15.85	17.14	17.03
Fbank SigRNN	18.23	15.40	**16.73**	16.79
Fbank LSTM	**17.05**	15.19	16.98	**16.41**
Combination by ROVER [39]	16.84	13.91	15.25	15.33

Table 2: *WER of self-trained fbank-DNN with 50-hours of supervised data and additional 190-hours of unsupervised data. FA and duration of unsupervised data after CM-based data selection are also listed.*

CM	FA	Dur.	WER (%)			
	(%)	(hour)	E1	E2	E3	E (ave)
(Baseline)	-	0	17.44	15.04	17.04	16.51
=1.0	96.0	90	17.16	14.62	16.66	16.15
>=0.9	93.8	129	17.08	14.61	16.79	16.16
>=0.8	92.8	140	**16.92**	**14.48**	16.65	16.02
>=0.7	91.9	148	16.98	14.61	16.49	16.03
>=0.0	86.5	190	17.03	14.64	**16.35**	**16.01**
(Oracle)	100.0	190	16.00	13.64	15.29	14.98

4-gram model with 644K sentences from the CSJ. Note that the data for LM training and data for {evaluation, development, unsupervised-training} did not overlap with each other. We used Kneser–Ney smoothing [38] and the vocabulary size was 77K. When decoding, we tuned the LM weight and word insertion penalty by using the development set. Then, the best LM weight and word insertion penalty of the development set was used to decode the evaluation sets.

4.2. Baseline 1: CE-model with small supervised data

We first evaluated the primary fbank-DNN and other four AMs. The results are listed in Table 1. Fbank-DNN was slightly better than MFCC and PLP-based DNNs. Fbank-based DNN, SigRNN and LSTM showed almost the same accuracies in this experiment. We believe the limited amount of training data made the accuracy differences between AMs small [2]. We could observe that the WERs for each evaluation set were very different for each AM although the five AMs produced similar WERs on average. In the rest of this paper, we focused on the semi-supervised training of the primary fbank-DNN AM.

4.3. Baseline 2: self-learning with CM thresholding

We then evaluated the self-learning method with CM-based data selection. We decoded the 190-hours of unsupervised data by using fbank-DNN AM. Then, reliable frames of the transcription were selected based on the CM. Finally, CE-training was conducted by using the combination of 50-hours of supervised data and selected portions of unsupervised data.

Results with various thresholding values are listed in Table 2. Note that if we used the higher thresholding values of CM (eg, 0.9), the total duration of the selected data became small while the frame accuracy (FA) of selected training samples became high. Thus, there is a trade-off between the accuracy and data size. Contrary to our expectation, the best WER (16.01%) was obtained when we used the training data with no thresholding (i.e., CM>=0.0), which corresponds to 3.0% of relative improvement.We thought that CM-based thresholding was not effective because the FA of transcription was relatively high in this experiment [3].

[2] In fact, we observed the better WER of SigRNN or LSTM than that of DNN with larger amount of training data in our preliminary experiments.

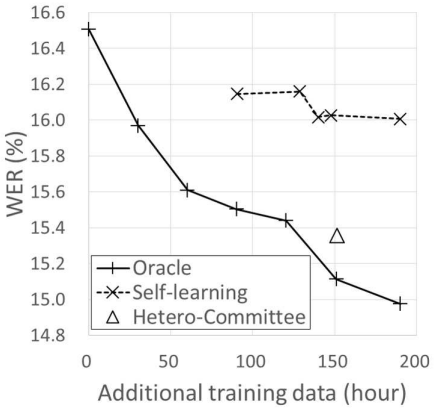

Figure 1: *Relation between WER and the amount of the additional unsupervised training data.*

It is important to note that only 0.36 points of improvement were obtained with CM-thresholding=1.0 despite the addition of 90 hours of highly accurate transcription (FA=96.0%). As a reference, we also trained AMs with a manual transcription of the unsupervised data (oracle settings). Results are shown in Figure 1 as "Oracle". As shown in this figure, if we could use only about 20 hours of oracle training data, we could achieve the same gain by the 90-hour data selected by CM. This result clearly demonstrated how inefficient the self-learning was. As mentioned in the introduction, this was because the data with high-CM was the data that the seed-AM had already been well-trained for.

When we used full manual transcription of 190 hours of unsupervised data, a much better WER of 14.98% was achieved, which corresponds to 9.3% of relative WER improvement from the baseline model. Compared to this oracle AM, the best self-trained DNN recovered only 32.7% of the oracle gain [4]. It is also noteworthy that the oracle result was better than the combination of five AMs by the ROVER method [39] (the last row of Table 1), which supported the importance of the semi-supervised training. These results led us to the investigation of the efficient semi-supervised training.

4.4. Semi-supervised training based on the committee of homogeneous AMs

We then evaluated the semi-supervised training based on the committee of three homogeneous AMs; fbank-DNN, MFCC-DNN and PLP-DNN. Results are listed in the upper part of Table 3. In this table, we also listed the FA and the duration of selected training samples from unsupervised data.

When we used the agreement parts of the three models, we obtained 145 hours of additional training samples with 91.8% of FA. Although these statistics were almost the same as the case of self-learning with threshold=0.7, slightly better WER of 15.94% was achieved. This result indicated that the committee-

[3] Note that CM thresholding was sometimes degraded the performance even in previous literatures like [13, 40].

[4] This measure is known as WER recovery [2], which we also use in the remaining sections.

Table 3: *WER of semi-supervised fbank-DNN based on the committee of homogeneous or heterogeneous networks. We used 50-hours of supervised data and additional 190-hours of unsupervised data. FA and duration of unsupervised data after data selection are also listed. Note that FA was calculated based on Fbank-DNN based alignment with the manual transcription.*

Committee	FA (%)	Dur (hour)	WER (%) E1	E2	E3	E (ave)	Relative Impr. (%)	WER Recovery (%) [2]
(Baseline)	-	0	17.44	15.04	17.04	16.51	-	-
Agreement of {Fbank, MFCC, PLP}-DNN	91.8	145	16.88	14.59	16.35	15.94	3.4	37.0
Agreement of {MFCC, PLP}-DNN	84.6	159	16.95	14.66	16.34	15.98	3.2	34.2
2/3 agreement of {Fbank, MFCC, PLP}-DNN	84.0	182	17.24	14.61	16.56	16.14	2.2	24.2
Agreement of {DNN, SigRNN, LSTM}	93.7	136	16.87	14.36	16.09	15.77	4.4	47.9
Agreement of {SigRNN, LSTM}	85.2	151	**16.61**	**14.17**	**15.29**	**15.36**	**7.0**	**75.2**
2/3 agreement of {DNN, SigRNN, LSTM}	84.1	179	16.75	14.26	15.65	15.55	5.8	62.3
(Oracle)	100	190	16.00	13.64	15.29	14.98	9.3	100.0

based data selection works as accurate as CM-based data selection, but with slightly high potential to find valuable training samples.

When we excluded the fbank-DNN from the committee ("Agreement of {MFCC, PLP}-DNN"), WER was marginally degraded. However, the important observation here is that while FA was much worse than the case of self-learning [5], the WER was comparable to that of self-learning. This result suggested that the FA is not necessarily the appropriate measure of training samples for semi-supervised training, as suggested in the self-learning experiment in which highly-accurate 90-hours data produced only marginal improvement. When we used the data in which two of three transcriptions agreed ("2/3 agreement of {Fbank, MFCC, PLP}-DNN"), the WER became much worse even than the self-learning. This result again suggested the unimportance of the primary-AM-based transcription.

4.5. Semi-supervised training based on the committee of heterogeneous AMs

Finally, we evaluated the semi-supervised training based on the committee of three heterogeneous AMs; fbank-DNN, fbank-SigRNN and fbank-LSTM. Results are listed in the lower part of Table 3. When we used the agreement part of the three models, 136 hours of training samples were selected with 93.7% of FA. The amount of training samples were smaller than the case of the committee of homogeneous networks (145 hours), which indicated that the models with different network structure had larger diversity than the models with different input feature set. Although the statistics were similar to the case of self-learning with threshold=0.9, the WER was much improved to 15.77% by adding this 136 hours of data.

The best, and the most interesting result was obtained when we excluded the primary fbank-DNN model from the committee ("agreement of SigRNN and LSTM"). In this case, the WER was further improved to 15.36%. Relative WER reduction became 7.0%, which corresponded to 75.2% of oracle gain by using full manual transcriptions. This phenomenon demonstrated that the primary-AM was not only useless but could be even harmful when constructing the training data for the primary-AM. Some readers may notice that the difference between this setting and the three-committee setting was only 15-hour data with very low FA (only 8%). Although the quality of the data seems poor, the data contained many "acceptable" labels with only few frames time-lag to the reference label created by fbank-DNN. In addition, the data was much valuable than other samples because the data was the one that the fbank-DNN never produced by itself. As a result, the large improvement was obtained regardless of the data size and its superficial quality. When we used the data in which two of three transcriptions agreed ("2/3 agreement of {DNN,SigRNN,LSTM}"), WER was degraded to 15.55%, which again indicated the unnecessity of the primary-AM when creating the data for the primary-AM.

[5]The FA could become worse than the worst case of self-learning (86.5%), because the agreement part of MFCC and PLP-based DNNs sometimes produced different labels that the fbank-DNN never produced.

Table 4: *Average WER of sMBR-trained fbank-DNN with various seed CE models. We used only 50-hours of supervised data for sMBR training.*

Seed CE-model of sMBR training	WER (%) Before sMBR	After sMBR
Baseline DNN with small data	16.51	15.93
Self-trained DNN	16.01	15.12
Hetero-committee-based DNN	**15.36**	**14.63**
Oracle DNN with full data	14.98	14.18

We plotted the best result in Figure 1. From this figure, we could see that while the improvements made by self-learning roughly correspond to only 30-hours of oracle data, the heterogeneous committee-based method achieved significant improvements which roughly correspond to adding 130-hours of oracle data. This result showed how efficient the heterogeneous committee-based method was compared to the self-learning approach.

Finally, we conducted the sMBR-training [41] starting from the semi-supervised-trained AMs. In this experiment, we used only 50-hours of supervised data because the sMBR-training was very sensitive to noise in automatic transcription and we could not achieve any improvement with semi-supervised settings. The results with various seed models are listed in Table 4. As shown in the table, the differences between semi-supervised methods were preserved even after sMBR training. Our best model based on the committee of heterogeneous models achieved 14.63% of WER after sMBR training, showing only 0.45 point difference with the oracle model-based result.

5. Conclusion

In this paper, we investigated the semi-supervised training in which additional complementary AMs are introduced to form a committee of label creation for untranscribed data. Especially, we investigated the case of using heterogeneous neural networks as complementary AMs, and the case of intentionally excluding the primary seed-AM from the committee. Both approaches achieved much better improvements than the self-learning or naive combination of homogeneous networks, showing 75.2% of recovery of oracle improvement.

6. References

[1] O. Chapelle, B. Schölkopf, and A. Zien, "Semi-supervised learning," *MIT press Cambridge*, 2006.

[2] J. Z. Ma and R. M. Schwartz, "Unsupervised versus supervised training of acoustic models." in *Proc. INTERSPEECH*, 2008, pp. 2374–2377.

[3] S. Novotney and R. Schwartz, "Analysis of low-resource acoustic model self-training," in *Proc. INTERSPEECH*, 2009.

[4] K. Vesely, M. Hannemann, and L. Burget, "Semi-supervised training of deep neural networks," in *Proc. ASRU*. IEEE, 2013, pp. 267–272.

[5] O. Siohan, "Training data selection based on context-dependent state matching," in *Proc. ICASSP*. IEEE, 2014, pp. 3316–3319.

[6] H. Xu, H. Su, E.-S. Chng, and H. Li, "Semi-supervised training for bottle-neck feature based dnn-hmm hybrid systems," in *Proc. INTERSPEECH*, 2014.

[7] F. Grézl and M. Karafiát, "Combination of multilingual and semi-supervised training for under-resourced languages," in *Proc. INTERSPEECH*, 2014.

[8] X. Cui, J. Huang, and J.-T. Chien, "Multi-view and multi-objective semi-supervised learning for HMM-based automatic speech recognition," *IEEE Trans. ASLP*, vol. 20, no. 7, pp. 1923–1935, 2012.

[9] T. Tsutaoka and K. Shinoda, "Acoustic model training using committee-based active and semi-supervised learning for speech recognition," in *Proc. APSIPA*. IEEE, 2012, pp. 1–4.

[10] F. Seide, G. Li, and D. Yu, "Conversational speech transcription using context-dependent deep neural networks." in *Proc. INTERSPEECH*, 2011, pp. 437–440.

[11] G. E. Dahl, D. Yu, L. Deng, and A. Acero, "Context-dependent pre-trained deep neural networks for large-vocabulary speech recognition," *IEEE Trans. SAP*, vol. 20, no. 1, pp. 30–42, 2012.

[12] R. Hsiao, T. Ng, F. Grézl, D. Karakos, S. Tsakalidis, L. Nguyen, and R. Schwartz, "Discriminative semi-supervised training for keyword search in low resource languages," in *Proc. ASRU*. IEEE, 2013, pp. 440–445.

[13] V. Manohar, D. Povey, and S. Khudanpur, "Semi-supervised maximum mutual information training of deep neural network acoustic models," in *Proc. INTERSPEECH*, 2015.

[14] S. Thomas, M. L. Seltzer, K. Church, and H. Hermansky, "Deep neural network features and semi-supervised training for low resource speech recognition," in *Proc. ICASSP*. IEEE, 2013, pp. 6704–6708.

[15] Y. Huang, D. Yu, Y. Gong, and C. Liu, "Semi-supervised GMM and DNN acoustic model training with multi-system combination and confidence re-calibration." in *Proc. INTERSPEECH*, 2013, pp. 2360–2364.

[16] J. Trmal, G. Chen, D. Povey, S. Khudanpur, P. Ghahremani, X. Zhang, V. Manohar, C. Liu, A. Jansen, D. Klakow *et al.*, "A keyword search system using open source software," in *Proc. SLT*. IEEE, 2014, pp. 530–535.

[17] S. Li, Y. Akita, and T. Kawahara, "Discriminative data selection for lightly supervised training of acoustic model using closed caption texts," in *Proc. INTERSPEECH*, 2015, pp. 3526–3530.

[18] L. Deng, O. Abdel-Hamid, and D. Yu, "A deep convolutional neural network using heterogeneous pooling for trading acoustic invariance with phonetic confusion," in *Proc. ICASSP*. IEEE, 2013, pp. 6669–6673.

[19] O. Vinyals, S. V. Ravuri, and D. Povey, "Revisiting recurrent neural networks for robust ASR," in *Proc. ICASSP*. IEEE, 2012, pp. 4085–4088.

[20] C. Weng, D. Yu, S. Watanabe, and B.-H. F. Juang, "Recurrent deep neural networks for robust speech recognition," in *Proc. ICASSP*. IEEE, 2014, pp. 5532–5536.

[21] G. Saon, H. Soltau, A. Emami, and M. Picheny, "Unfolded recurrent neural networks for speech recognition," in *Proc. INTERSPEECH*, 2014.

[22] N. Kanda, M. Tachimori, X. Lu, and H. Kawai, "Training data pseudo-shuffling and direct decoding framework for recurrent neural network based acoustic modeling," in *Proc. ASRU*, 2015, pp. 15–21.

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

[24] H. Sak, A. Senior, and F. Beaufays, "Long short-term memory based recurrent neural network architectures for large vocabulary speech recognition," *arXiv e-prints*, 2014.

[25] H. Sak, O. Vinyals, G. Heigold, A. Senior, E. McDermott, R. Monga, and M. Mao, "Sequence discriminative distributed training of long short-term memory recurrent neural networks," *entropy*, vol. 15, no. 16, pp. 17–18, 2014.

[26] J. T. Geiger, Z. Zhang, F. Weninger, B. Schuller, and G. Rigoll, "Robust speech recognition using long short-term memory recurrent neural networks for hybrid acoustic modelling," in *Proc. INTERSPEECH*, 2014.

[27] H. Sak, A. Senior, and F. Beaufays, "Long short-term memory recurrent neural network architectures for large scale acoustic modeling," in *Proc. INTERSPEECH*, 2014.

[28] H. Soltau, B. Kingsbury, L. Mangu, D. Povey, G. Saon, and G. Zweig, "The ibm 2004 conversational telephony system for rich transcription." in *ICASSP (1)*, 2005, pp. 205–208.

[29] S. F. Chen, B. Kingsbury, L. Mangu, D. Povey, G. Saon, H. Soltau, and G. Zweig, "Advances in speech transcription at ibm under the darpa ears program," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 14, no. 5, pp. 1596–1608, 2006.

[30] T. Shinozaki, Y. Kubota, and S. Furui, "Unsupervised acoustic model adaptation based on ensemble methods," *Selected Topics in Signal Processing, IEEE Journal of*, vol. 4, no. 6, pp. 1007–1015, 2010.

[31] J. L. Elman, "Finding structure in time," *Cognitive science*, vol. 14, no. 2, pp. 179–211, 1990.

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

[33] K. Maekawa, "Corpus of spontaneous japanese: Its design and evaluation," in *ISCA & IEEE Workshop on Spontaneous Speech Processing and Recognition*, 2003.

[34] N. Kanda, R. Takeda, and Y. Obuchi, "Elastic spectral distortion for low resource speech recognition with deep neural networks," in *Proc. ASRU*, 2013, pp. 309–314.

[35] Y. Tachioka and S. Watanabe, "A discriminative method for recurrent neural network language models," in *Proc. ICASSP*, 2015.

[36] F. Seide, G. Li, X. Chen, and D. Yu, "Feature engineering in context-dependent deep neural networks for conversational speech transcription," in *Proc. ASRU*. IEEE, 2011, pp. 24–29.

[37] J. Martens, "Deep learning via hessian-free optimization," in *Proc. ICML*, 2010, pp. 735–742.

[38] S. F. Chen and J. Goodman, "An empirical study of smoothing techniques for language modeling," *Computer Speech & Language*, vol. 13, no. 4, pp. 359–393, 1999.

[39] J. G. Fiscus, "A post-processing system to yield reduced word error rates: Recognizer output voting error reduction (rover)," in *Proc. ASRU*. IEEE, 1997, pp. 347–354.

[40] H. Su and H. Xu, "Multi-softmax deep neural network for semi-supervised training," in *Proc. INTERSPEECH*, 2015.

[41] K. Veselỳ, A. Ghoshal, L. Burget, and D. Povey, "Sequence-discriminative training of deep neural networks." in *Proc. INTERSPEECH*, 2013, pp. 2345–2349.