Distilling Knowledge for Distant Speech Recognition via Parallel Data

Jiangyan Yi[1,3] and Jianhua Tao[1,2,3]
[1]National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China
[2]CAS Center for EBSIT, Institute of Automation, Chinese Academy of Sciences, Beijing, China
[3] University of Chinese Academy of Sciences, Beijing, China
E-mail: {jiangyan.yi, jhtao}@nlpr.ia.ac.cn

Abstract— **In order to improve the performance of distant speech recognition tasks, this paper proposes to distill knowledge from the close-talking model to the distant model using parallel data. The close-talking model is called the teacher model. The distant model is called the student model. The student model is trained to imitate the output distributions of the teacher model. This constraint can be realized by minimizing the Kullback-Leibler (KL) divergence between the output distribution of the student model and the teacher model. Experimental results on AMI datasets show that the best student model achieves up to 8.5% relative word error rate (WER) reduction when compared with the conventionally-trained baseline models.**

I. Introduction

In a close-talking setting, automatic speech recognition systems have achieved significant improvement with deep neural network (DNN) based acoustic models [1, 2, 3]. However, distant speech recognition tasks are still challenging [4], especially when dealing with speech collected from a single distant microphone.

A lot of efforts have been made to improve the performance of distant speech recognition systems [5, 6, 7]. Many of these approaches use time-synchronize close-talking and distant parallel data [8, 9, 10].

Some literatures utilize the close-talking data together with the distant data to train acoustic models for speech recognition. One of the methods is the multi-condition training [10, 11]. This method just uses all the data from different conditions to train acoustic models. The other method is environment-aware training [6, 12]. This approach has been proposed to use close-talking features to help extract environment features as auxiliary information. Other works are proposed to use the enhanced speech to train acoustic models for speech recognition. The dereverberation model is used to estimate the close-talking data given the distant data [13, 14]. Some researchers [14, 15] train the dereverberation model and the recognition model independently. Others [16, 17, 18, 19] propose a joint training approach between speech enhancement and speech recognition tasks. Moreover, Ravanelli et al. [20] propose a novel network where speech enhancement and speech recognition tasks cooperate with each other. Most recently, Heymann et al. [21] and Menne et al. [22] propose to jointly optimize the enhanced model and the acoustic model.

The above-mentioned approaches are able to obtain obvious improvement. However, most of them only use the close-talking data as the training data or the optimized reference. Few of them use the close-talking model to guide the training of the distant model. More recently, Qian et al. in [10] propose to share knowledge between two hidden layers of the close-talking and the distant models. This approach achieves promising improvement. However, it only shares knowledge between the hidden layers rather than transfer knowledge between the output layers of the two models.

Therefore, knowledge distillation is proposed to transfer knowledge between the output layers of the close-talking and the distant models in this paper. The concept of knowledge distillation has been around for a decade [23, 24]. A more general framework is proposed by Hinton et al. [25] to distill knowledge by using high temperature. At a high level, distillation contains training a new model. The new model is trained to mimic the output distribution of a well-trained model.

Similarly, there are several works that use knowledge distillation to compress acoustic models. Li et al. [26] utilize a large DNN model to train a small DNN model. In [27], Chan et al. propose to transfer knowledge from a recurrent neural networks (RNN) model to a small DNN model. Chebotar et al. [28] propose to distill ensembles of acoustic models into a single acoustic model. All of these methods utilize Kullback-Leibler (KL) divergence [29] to minimize the difference of the output distributions between the two acoustic models. Previous results show that these methods can compress acoustic models effectively with a little performance loss.

Inspired by the above methods, this paper uses KL divergence to distill knowledge using parallel data to improve the performance of the distant speech recognition task. An acoustic model trained with the close-talking data is called a teacher model. An acoustic model trained with the distant data is called a student model. The student model is trained to imitate the output distribution of the teacher model. The difference between the output distributions of the two models can be minimized by KL divergence. In addition, this paper investigates how the improvement of the student model is influenced by the performance of the teacher models.

Experimental results on AMI corpus [30] show that the best student model achieves up to 8.5% relative word error rate (WER) reduction when compared with the conventionally-trained baseline models. The results also show that increases in the accuracy of the teacher model yield similar increases in the performance of the student model.

The rest of this paper is organized as follows. Section 2 describes knowledge distillation using parallel data. Experiments are presented in Section 3. The results are discussed in Section 4. This paper is concluded in Section 5.

II. KNOWLEDGE DISTILLATION VIA PARALLEL DATA

In this section, the algorithm of distillation is introduced at first. Then the framework of knowledge distillation for distant speech recognition is presented in detail.

A. Distillation

The distillation is to make the teacher model transfer knowledge to the student model. The student model is trained to mimic the output distribution of the teacher model. Thus, the student model is forced to be close to the output distribution of the teacher model. This constraint can be realized by minimizing the KL divergence between the output distributions of the two models. Letting P_c denotes the output probabilities of the teacher model, Q_f denotes the output probabilities of the student model, the difference of the output distributions between the two models is defined as $D_{KL}(P_c||Q_f)$ which is wished to minimize.

$$D_{KL}\left(P_c||Q_f\right) = \sum_i P_c(s_i|x_c) ln\,(P_c(s_i|x_c)/Q_f(s_i|x_f)) \quad (1)$$

where i denotes the index of a senone, s_i denotes the i-th senone, x_c is referred as input features of the close-talking speech, x_f is referred as input features of the far-field speech, $Q_f(s_i|x_f)$ denotes the posterior probability of s_i computed from the student model given x_f , $P_c(s_i|x_c)$ denotes the posterior probability of s_i computed from the teacher model given x_c. $D_{KL}\left(P_c||Q_f\right)$ can be also defined as follows.

$$D_{KL}\left(P_c||Q_f\right) = H\left(P_c, Q_f\right) - H(P_c) \quad (2)$$

$$H\left(P_c, Q_f\right) = \sum_i -P_c(s_i|x_c) ln Q_f(s_i|x_f) \quad (3)$$

$$H(P_c) = \sum_i -P_c(s_i|x_c)\, ln P_c(s_i|x_c) \quad (4)$$

Equation (4) is only correlated with the teacher model. So, Equation (4) can be neglected. Thus, we can define

$$D_{KL}\left(P_c||Q_f\right) \triangleq \sum_i -P_c(s_i|x_c) ln Q_f(s_i|x_f) \quad (5)$$

By Equation (5), we can see that the KL divergence is minimized by minimizing the cross entropy (CE) loss function. Thus, the optimization of the distillation can be viewed as the standard CE training criterion. Therefore, the normal backpropagation (BP) algorithm can be directly used to train the student model. The only thing that needs to be changed is that the hard label t_{hard} is replaced with $P_c(s_i|x_c)$. $P_c(s_i|x_c)$ is called soft label t_{soft}.

Equation (5) also indicates that we can still transfer knowledge from the teacher model to the student model, if the loss function or network architecture of the student model is different from the teacher model. It only needs that the output labels of the student model are identical to the teacher model. This approach is a simplified version of the high temperature based distillation proposed by Hinton et al. [25].

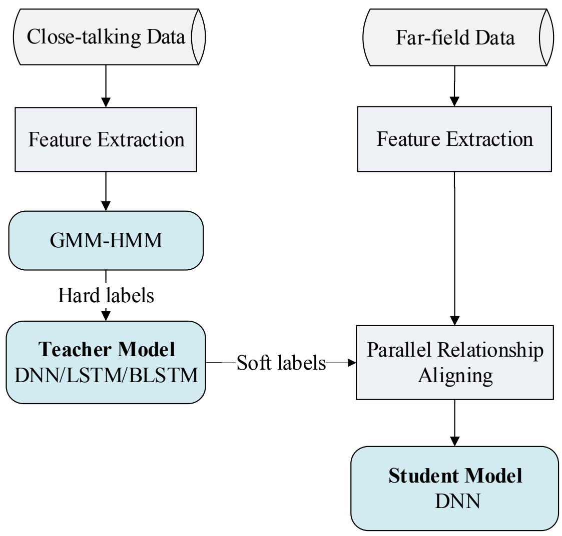

Figure 1: *Framework of knowledge distillation for distant speech recognition.*

B. Framework of knowledge distillation

The framework of knowledge distillation for far-field speech recognition is shown in Fig. 1. The training of the student model is guided by the teacher model using parallel data. The teacher models and the student models are hybrid acoustic models. They have identical output labels which are senones.

The hard labels t_{hard} are generated from the Gaussian mixture model hidden Markov model (GMM-HMM) model by frame-level. The GMM-HMM model is trained with the close-talking data. The hard labels are one-hot vectors. For example, [0 0 0 1 0 0] denotes the hard labels of one frame. The probability of this frame belonging to label 4 is 1. The probability of this frame belonging to other labels is 0.

The teacher model is trained with the close-talking data x_c and the above hard labels t_{hard}. The neural network of the teacher model can be based on DNN [1], long short term memory (LSTM) [31] or bidirectional LSTM (BLSTM). After the training, the parameters of the teacher model are fixed. The teacher model is only used to compute the soft labels.

The soft labels t_{soft} are computed from the teacher model using forward algorithm with the close-talking data x_c by frame-level. The soft labels have much more information about underlying label distribution than the hard labels. For example, [0.01 0.1 0.03 0.79 0 0.07] denotes the soft labels of one frame. The probability of this frame belonging to label 4 is 0.79. The probability of this frame belonging to label 1 is 0.01.

The student model is only DNN based acoustic model. The parallel relationship is used to align the far-field data x and the close-talking soft labels t_{soft}. Then the student model is trained using far-field data x with the corresponding soft labels. The training criterion is Equation (5). The parameters of the student model are updated but the parameters of the teacher model aren't changed, when training the student model.

At the decoding stage, only the student model is used to compute posterior probabilities. Then the acoustic likelihood can be computed by combining posterior with prior probabilities. Thus, our proposed method doesn't need extra computation cost for decoding.

III. Experiments

A. Datasets

Our experiments are conducted on AMI Meeting Corpus [30]. This corpus consists of 100 hours of meeting recordings. The recordings use a range of signals synchronized to a common timeline. There are three types of recordings: individual headset microphones (IHM), single distant microphone (SDM) and multiple distant microphones (MDM) datasets. IHM is the close-talking data which is collected from individual headset microphones. SDM is the distant data which is collected from a single distant microphone using 1st microphone array. MDM is the distant data which is collected from multiple distant microphones using multiple microphones array. Our experiments only use IHM and SDM datasets. There are three sets for the IHM and SDM datasets respectively: training set (*train*), development set (*dev*) and test set (*eval*). The training set contains 108221 utterances about 80 hours. The development set has 13059 utterances about 10 hours. The test set has 12612 utterances about 10 hours.

B. Experimental setup

The proposed approach is implemented based on Kaldi speech recognition toolkit [32]. In order to compare our proposed method with the methods in [10], we follow the experimental setup in [10]. The frame length is 25ms and the frame shift is 10ms. The input features of all GMM-HMM models are 39-dim MFCC features. The models have 80K Gaussians. The input features of all neural networks are 40-dimensional log mel-filter bank (FBANK) features plus delta and delta-delta. The parameters of all the models are updated on the train set. The training terminates on the dev set with a little improvement. The dev set is also used to adjust the hyper parameters and select the models. The vocabulary is from the AMI dictionary which has 50K words. The language model (LM) is a trigram. The LM is trained using the AMI training transcripts and the Fisher English corpus. The decoding procedure is followed the standard AMI recipe.

C. Baseline model

We follow the officially released Kaldi recipe to build two GMM-HMM models at first. The Distant-GMM is trained with the distant data. The Close-GMM is trained with the close-talking data. The Distant-GMM has 4237 senones. The Close-GMM has 4239 senones.

Then we use the distant data from the SDM dataset to train two DNN models. One is called the Distant-DistantAli-DNN which is trained with the hard labels generated from the Distant-GMM using the SDM dataset. The other is called Distant-CloseAli-DNN which is trained with the hard labels generated from the Close-GMM using the IHM dataset.

Table 1: *WER (%) of three models on the SDM dataset.*

Model	**Hard labels**	**Dev**	**Eval**
Distant-GMM	-	64.4	69.5
Distant-DistantAli-DNN	SDM	54.0	58.6
Distant-CloseAli-DNN	IHM	**50.6**	**55.4**

The two DNN models have 6 hidden layers with 2048 sigmoid units in each layer. The input layer of the models uses a sliding context window of 11 frames. The models are trained using the stochastic gradient descent (SGD) with mini-batch size of 256. The initial learning rate is set to 1×10^{-3}. The results of the Distant-GMM model and the two DNN models on *dev* and *eval* sets of the SDM dataset are listed in Table 1.

The results in Table 1 show that the Distant-CloseAli-DNN model outperforms other models on *dev* and *eval* sets obviously. The results show that the use of close-talking hard labels leads to obvious improvement. The reason is that the close-talking hard labels have higher quality than the distant hard labels.

The results are consistent with the conclusions in [10, 33]. Therefore, we select the strongest Distant-CloseAli-DNN as the baseline model to compare with our student models.

D. The effect of the close-talking teacher models

In order to investigate how the improvement of the student model is influenced by the teacher models, all the student models have the same architecture configuration. The student models are DNN based acoustic models which have the same configuration to the baseline Distant-CloseAli-DNN model.

There are four teacher models trained using close-talking data from the IHM dataset. The hard labels are generated from the Close-GMM model using the IHM dataset for all the teacher models.

Close-DNN-Teacher: the model has the same number of parameters with the baseline Distant-CloseAli-DNN model.

Close-DNN-sMBR-Teacher: the model is the above Close-DNN-Teacher model retrained with state-level minimum Bayes risk (sMBR). This model is iterated by 2 epoches.

Close-LSTM-Teacher: the model uses a single frame as input. It has 4 stacked LSTM layers with projection, and each layer has 1024 memory cells and 512 output units. The initial learning rate and momentum are set to 0.0001 and 0.9 respectively. The training is carried out by truncated BP through time (BPTT) algorithm.

Close-BLSTM-Teacher: the model uses a single frame as input. It has 4 stacked BLSTM layers with projection, and each layer has 512 memory cells and 256 output units. The initial learning rate and momentum are set to 0.0001 and 0.9 respectively. The training is carried out by BPTT algorithm.

There are four student models trained using distant data from the SDM dataset: Distant-DNN-Student 1-4. The teacher model of the Distant-DNN-Student 1 is the Close-DNN-Teacher model. The Distant-DNN-Student 2 is trained to mimic the Close-DNN-sMBR-Teacher model.

Table 2: *WER (%) of the teacher models on the IHM dataset and student models on the SDM dataset.*

Models	Dev	Eval
Close-GMM	32.2	35.1
Distant-CloseAli-DNN	50.6	55.4
Close-DNN-Teacher	27.1	28.2
Distant-DNN-Student 1	47.4	53.3
Close-DNN-sMBR-Teacher	26.0	26.1
Distant-DNN-Student 2	46.1	52.1
Close-LSTM-Teacher	24.2	24.7
Distant-DNN-Student 3	45.6	51.9
Close-BLSTM-Teacher	**22.5**	**22.8**
Distant-DNN-Student 4	**44.5**	**50.7**

The teacher model of the Distant-DNN-Student 3 is the Close-LSTM-Teacher model. The Distant-DNN-Student 4 is guided by the DNN-BLSTM teacher model. The soft labels are computed from the teacher models using the IHM dataset respectively.

The results of these models are listed in Table 2. The results in Table 2 show that the student model will achieve better performance, if the teacher model has higher accuracy. The Close-BLSTM-Teacher model achieves the best performance. Therefore, The Distant-DNN-Student 4 model outperforms all other student models both on the *dev* and *test* dataset.

E. Distant student model

We also compare our proposed method with other methods. DRSL is the method proposed in [14], Multi-Cond, DRJL-Parallel, DRJL-Front-Back, CFMKS and DRJL+CFMKS are the approaches proposed in [10]. We implement these methods following the configuration in [10, 14].

DRSL: training the dereverberation and speech recognition models independently. Multi-Cond: just directly using all the data from close-talking and distant to train acoustic models.DRJL-Parallel: joint training between the dereverberation and speech recognition models sharing hidden layers. DRJL-Front-Back: joint training between the dereverberation and speech recognition models in front-back structure. CFMKS: sharing knowledge between two hidden layers in the close-talking and the distant models. All of these models are DNN based. For Multi-Cond, DRJL-Parallel, DRJL-Front-Back, CFMKS, the models are trained using 6 hidden layers with 2048 sigmoid units in each layer. For DRSL and DRJL-Front-Back, both the dereverberation and recognition models are trained using 3 hidden layers with 2048 sigmoid units in each layer respectively.

The results are evaluated on the *dev* and *eval* sets of the SDM dataset are listed in Table 3. The results in Table 3 show that our best student model Distant-DNN-Student 4 outperforms all other models. Our best student model achieves 8.5% relative WER reduction on *eval* set when compared with the baseline model, and obtains 4.0% relative WER reduction on *eval* set over the best model DRJL+CFMKS. Our best student model outperforms the CFMKS model by 5.8% relative WER reduction, and also outperforms the DRJL-Front-Back model by 5.4% relative WER reduction on eval set.

Table 3: *The WER (%) of the best student model and other models evaluated on the SDM dataset.*

Model	Dev	Eval
Distant-CloseAli-DNN (Baseline)	50.6	55.4
DRSL	49.8	54.8
Multi-Cond	50.1	54.9
DRJL-Parallel	48.8	54.2
DRJL-Front-Back	47.9	53.6
CFMKS	48.1	53.8
DRJL+CFMKS	47.0	52.8
Distant-DNN-Student 4 (Ours)	**44.5**	**50.7**

The results also show that Multi-Cond and DRSL can only obtain a small gain over the baseline model. DRJL-Front-Back achieves more improvement than DRSL.

IV. Discussion

The above experiments show that our proposed method is effective. Some interesting observations are made as follows.

The best student model outperforms the baseline model and the other conventionally-trained models. There are two main reasons. One is that the teacher model can capture more accurate and better phoneme features from the close-talking data. In contrast, some of the phoneme features from the distant data are distorted by reverberation and noise. The other is that the soft labels computed from the teacher model contain more information about underlying label distributions when compared with the hard labels. Thus, the student model is easier to learn well using more accurate and richer information.

Our student model outperforms the CFMKS model. The main reason is that the output layers have stronger discriminative ability than the hidden layers. The CFMKS method only shares knowledge between hidden layers. But our proposed method transfers knowledge between output layers.

Our student model also outperforms the DRJL-Front-Back model. One possible explanation is that the speech may be distorted by the dereverberation model. Nevertheless, the student model only changes its output distribution to imitate the strong teacher model.

In addition, increases in the accuracy of the teacher models yield similar increases in the performance of the student model. If the teacher model has higher accuracy, the student model can train well using more accurate soft labels.

V. Conclusions

This paper proposes to distill knowledge from the teacher model to the student model using parallel data to improve the performance of distant speech recognition tasks. The student model is trained to mimic the output distribution of the teacher model. Thus, it can be realized by minimizing the KL divergence between the output distributions of the two models. Experimental results on AMI corpus show that the best student model achieves up to 8.5% relative WER reduction when compared with the conventionally-trained baseline models. The results also show that increases in the accuracy of the teacher model yield similar increases in the performance of the student model. Moreover, our proposed method doesn't need extra computation cost for decoding. In future work, we plan to use the ensemble teacher model to improve the performance of the student model and apply this approach to other tasks.

ACKNOWLEDGMENT

This work is supported by the National Key Research & Development Plan of China (No.2018YFB1005003), the National Natural Science Foundation of China (NSFC) (No.61425017, No.61831022, No.61771472, No.61603390), the Strategic Priority Research Program of Chinese Academy of Sciences (No.XDC02050100), and Inria-CAS Joint Research Project (No.173211KYSB20170061).

REFERENCES

[1] G. Hinton, L. Deng, D. Yu, G. Dahl, A. Mohamed, N. Jaitly, A. Senior, V. Vanhoucke, P. Nguyen, and T. NSainath, “Deep neural networks for acoustic modeling in speech recognition: The shared views of four research groups,” Signal Processing Magazine, IEEE, vol. 29, no. 6, pp. 82–97, 2012.

[2] W. Xiong, J. Droppo, X. Huang, F. Seide, M. Seltzer, and A.Stolcke, “The Microsoft 2016 conversational speech recognition system, ” in Proceedings of ICASSP, 2017.

[3] G. Saon, T. Sercu, S. Rennie, and H. K. J. Kuo, “The IBM 2016 english conversational telephone speech recognition system,” in Proceedings of Interspeech, 2016.

[4] T. Hain, L. Burget, J. Dines, P. N. Garner, F. Grezl and A. E. Hannani, “Transcribing meetings with the amida systems, ” IEEE Transactions on Audio, Speech, and Language Processing, vol. 20, no. 2, pp. 486–498, 2012.

[5] P. Swietojanski, A. Ghoshal, and S. Renals, “Hybrid acoustic models for distant and multichannel large vocabulary speech recognition,” in Proceedings of ASRU, 2013, pp. 285–290.

[6] R. Giri, M. L Seltzer, J. Droppo, and D. Yu, “Improving speech recognition in reverberation using a room aware deep neural network and multi-task learning,” in Proceedings of ICASSP, 2015, pp. 5014–5018.

[7] S. Kim and I. Lane, “Recurrent models for auditory attention in multi-microphone distance speech recognition,” in Proceedings of Interspeech, 2016, pp. 3839-3842.

[8] K. Kumatani and J. McDonough, “Microphone array processing for distant speech recognition: from close-talking microphones to far-field sensors,” Signal Processing Magazine, IEEE, vol. 29, no. 6, pp. 127–140, 2012.

[9] F. Weninger, H. Erdogan, S. Watanabe, E. Vincent, J. Le Roux, J. R. Hershey, and B. W. Schuller, “Speech enhancement with LSTM recurrent neural networks and its application to noise robust ASR,” in Proceedings of LVA/ICA, 2015, pp. 91–99.

[10] Y. Qian, T. Tan, and D. Yu, “An investigation into using parallel data for far-field speech recognition,” in Proceedings of ICASSP, 2016, pp. 5725–5729.

[11] I. Himawan, P. Motlicek, D. Imseng, B. Potard, N. Kim, and J. Lee, “Learning feature mapping using deep neural network bottleneck features for distant large vocabulary speech recognition,” in Proceedings of ICASSP, 2015, pp. 4540–4544.

[12] M. L Seltzer, D. Yu, and Y. Wang, “An investigation of deep neural networks for noise robust speech recognition,” in Proceedings of ICASSP, 2013, pp. 7398–7402.

[13] Y. Xu, J. Du, L. R. Dai, and C. H. Lee, “A regression approach to speech enhancement based on deep neural networks,” IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 23, no. 1, pp. 7–19, Jan 2015.

[14] J. Du, Q. Wang, T. Gao, Y. Xu, L. Dai, and C. H. Lee, “Robust speech recognition with speech enhanced deep neural networks,” in Proceedings of Interspeech, 2014, pp. 616–620.

[15] M. Mimura, S. Sakai, and T. Kawahara, “Deep autoencoders augmented with phone-class feature for reverberant speech recognition,” in Proceedings of ICASSP, 2015, pp. 4365–4369.

[16] T. Gao, J. Du, L. R. Dai, and C. H. Lee, “Joint training of front-end and back-end deep neural networks for robust speech recognition,” in Proceedings of ICASSP, 2015, pp. 4375–4379.

[17] Z. Chen, S. Watanabe, H. Erdogan, and J. Hershey, “Speech enhancement and recognition using multi-task learning of long short-term memory recurrent neural networks,” in Proceedings of Interspeech, 2015, pp. 3274–3278.

[18] M. Mimura, S. Sakai, and T. Kawahara, “Joint optimization of denoising autoencoder and dnn acoustic model based on multitarget learning for noisy speech recognition,” in Proceedings of Interspeech, 2016.

[19] T. Higuchi, T. Yoshioka, and T. Nakatani, “Optimization of speech enhancement front-end with speech recognition-level criterion,” in Proceedings of Interspeech, 2016.

[20] M. Ravanelli, P. Brakel, M. Omologo and Y. Bengio, “A network of deep neural networks for distant speech recognition,” in Proceedings of ICASSP, 2017, pp.4880-4884.

[21] J. Heymann, L. Drude, R. Haeb-Umbach, K. Kinoshita, T. Nakatani, “Joint Optimization of Neuralnet Work-Based Wpe Dereverberation and Acoustic Model for Robust Online ASR”, in Proceedings of ICASSP, 2019, pp.6655-6659.

[22] T. Menne, R. Schluter, H. Ney, “Investigation into Joint Optimization of Single Channel Speech Enhancement and Acoustic Modeling for Robust ASR”, in Proceedings of ICASSP, 2019, pp.6655-6659.

[23] C. Bucila, R. Caruana, and A. Niculescu-Mizil, “Model compression,” in ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2006.

[24] J. Ba and R. Caruana, “Do deep nets really need to be deep?” in Advances in Neural Information Processing Systems, 2014, pp. 2654–2662.

[25] G. Hinton, O. Vinyals, and J. Dean, “Distilling the knowledge in a neural network,” in Neural Information Processing Systems: Workshop Deep Learning and Representation Learning Workshop, 2014.

[26] J. Li, R. Zhao, J. T. Huang, and Y. Gong, “Learning small-size dnn with output-distribution-based criteria,” in Proceedings of Interspeech, 2014.

[27] W. Chan, N.R. Ke, and I. Lane, “Transferring knowledge from a RNN to a DNN,” in Proceedings of Interspeech, 2015.

[28] Y. Chebotar and A. Waters, “Distilling knowledge from ensembles of neural networks for speech recognition,” in Proceedings of Interspeech, 2016, pp. 3439-3443.

[29] S. Kullback and R. A. Leibler, “On information and sufficiency,” Ann. Math. Statist., vol. 22, no. 1, pp. 79–86, 1951.

[30] I. McCowan, J. Carletta, W. Kraaij, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, and V. Karaiskos, “The ami meeting corpus,” in Proceedings of the 5th International

Conference on Methods and Techniques in Behavioral Research, vol. 88, 2005.
[31] H. Sak, A. Senior, and F. Beaufays, “Long Short-Term Memory Based Recurrent Neural Network Architectures for Large Vocabulary Speech Recognition,” in the Proceedings of Interspeech, 2014.
[32] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Qian, Y.M., P. Schwarz, J. Silovsky, G. Stemmer, and K. Vesely, “The Kaldi speechrecognition toolkit,” in the Proceedings of ASRU, 2011.
[33] V. Peddinti, V. Manohar, Y. Wang, D. Povey, and S. Khudanpur, “Far-field ASR without parallel data,” in Proceedings of Interspeech, 2016, pp. 1996-2000.