FULLY LEARNABLE FRONT-END FOR MULTI-CHANNEL ACOUSTIC MODELING USING SEMI-SUPERVISED LEARNING

Sanna Wager[1*], *Aparna Khare*[2], *Minhua Wu*[2]
Kenichi Kumatani[2], *Shiva Sundaram*[2]

[1] Indiana University, School of Informatics, Computing, and Engineering, Bloomington, IN, USA
[2]Amazon, Inc, Sunnyvale, CA

ABSTRACT

In this work, we investigated the teacher-student training paradigm to train a fully learnable multi-channel acoustic model for far-field automatic speech recognition (ASR). Using a large offline teacher model trained on beamformed audio, we trained a simpler multi-channel student acoustic model used in the speech recognition system. For the student, both multi-channel feature extraction layers and the higher classification layers were jointly trained using the logits from the teacher model. In our experiments, compared to a baseline model trained on about 600 hours of transcribed data, a relative word-error rate (WER) reduction of about 27.3% was achieved when using an additional 1800 hours of untranscribed data. We also investigated the benefit of pre-training the multi-channel front end to output the beamformed log-mel filter bank energies (LFBE) using L2 loss. We find that pre-training improves the word error rate by 10.7% when compared to a multi-channel model directly initialized with a beamformer and mel-filter bank coefficients for the front end. Finally, combining pre-training and teacher-student training produces a WER reduction of 31% compared to our baseline.

Index Terms— far-field automatic speech recognition, acoustic modeling, knowledge distillation, teacher-student training

1. INTRODUCTION

Multi-channel far-field automatic speech recognition systems typically perform multiple tasks, including voice activity detection, speaker localization, speech enhancement, beamforming, and acoustic modeling. Beamforming is an important component at the front-end that uses spatial information from a microphone array to improve robustness against noise or reverberation, which can improve ASR accuracy [1]. In traditional approaches [2, 3, 4, 5, 6, 7], beamforming is performed as a pre-processing step, and is not data-driven: its output is used as input to the trainable part of the acoustic model (AM). While this approach is effective, it can often fail when the room and speaker conditions do not match the design criteria. Prior work has shown that learning a multi-channel front-end jointly with the AM using the ASR objective can improve far-field performances. In [8], Sainath et al. showed that input from a data-driven multi-channel front-end provides better results than both single-channel and beamformed input. They introduce a set of convolutional filters applied directly to the raw audio [8]. The convolutional and linear structures are both designed to explicitly incorporate multiple beamformer "look directions", subsuming a multi-geometry beamforming component into the deep neural network (DNN). Ochiai et al. present a bi-directional LSTM structure that learns either a filter or mask estimation beamformer given frequency-domain input signals [9]. Wu et al. in [10] use a set of linear transformations applied to the frequency-domain input signals that also subsumed the notion of look directions. That work primarily introduced initializing the spatial filtering layers with superdirective beamformer coefficients and training the model in a stage-wise manner. This was also extended to the multi-geometry case by Kumatani et al. in [11]. The multi-channel spatial filtering approach described in these works allows the front-end to be trained on challenging real-world examples directly on the ASR task.

Teacher-student training (T/S) or knowledge distillation was described in [12] and [13]. The authors demonstrated that the posterior probabilities generated by powerful offline "teacher" models can be used to train simpler "student" models and the technique was successfully applied to the acoustic modeling problem in [12]. This technique has also been successfully applied for domain adaption for ASR. In Li et al. [14], the authors improve speech recognition performance of a distant microphone by applying T/S training to utterances recorded simultaneously using a close-talking distant microphones. In a similar vein, Mosner et al. [15] apply T/S to improve noise robustness by creating a parallel corpus by adding multimedia interference to clean utterances. T/S strategy has also been used for improving the overall ASR performance of the student model by leveraging significantly larger amount of untranscribed or unlabelled speech data. Parthasarathi et al. [16] present results on using up to 1 million hours of untranscribed data.

*Sanna Wager performed this work as a research intern at Amazon.

In contrast to the prior work listed above, the main contribution of this work is the use of teacher-student training to jointly train the spatial filtering, feature extraction and classification layers. It combines and builds upon the prior work where we train a unified multi-channel acoustic model by leveraging significantly larger amount of untranscribed data for acoustic model training. In contrast to other works that used teacher-student training for domain adaptation or knowledge distillation, we specifically focus on learning a front-end and improve the performance of a multi-channel ASR by training the model on real-world examples. We also experimentally evaluate the benefit of pre-initializing the front-end layers. Specifically, we compare initializing the spatial filtering weights with traditional signal processing based beamformer coefficients against a data-driven approach that uses L2-loss.

This paper is structured as follows. Section 2 introduces the network architecture used in our experiments along with initialization, pre-training approaches, and the T/S training. Section 3 describes the datasets and training techniques we use for our experiments, our experimental setup and the results. Finally, Section 4 concludes the paper.

2. MODEL STRUCTURE

2.1. Multi-channel acoustic model

The model architecture used in our experiments is the *elastic spatial filtering* used in [10]. We compute a 128-dimensional discrete Fourier transform (DFT) over a 12.5-ms window with a 10-ms frame shift, removing the DC component. After global mean-variance normalization (GMV norm.), the channels are input to the front-end component of the network, which is the combination of a beamformer and a LFBE feature extractor. Next is a complex power operation reducing the dimension to 1536, followed by a linear layer that produces a weighted combination of the spatial filtering layer outputs and reduces the dimension to 127. The feature extraction component consists of a Mel filter Bank (MFB) layer with output dimension 64, followed by the ReLU and log operations, which mimic the LFBE. Note that the various linear layers are named after their expected digital signal processing (DSP) functionality but, with training, might learn different transformations.

The classification component of the AM contains 5 unidirectional LSTM layers with a hidden dimension of 768, followed by a linear layer and the softmax operation that outputs a 3183-dimensional senone probability distribution. The total number of weights in the network is 29.6M. The model structure is shown on the right side of Figure 2 that describes the overall T/S training. The model architecture is consistent across all experiments described in this paper.

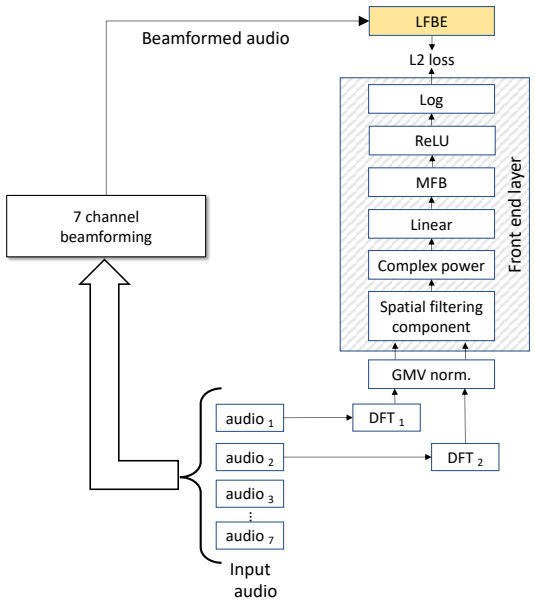

Fig. 1. Pre-training of the front-end layers up to the classification network. LFBE targets are computed using beamformed audio using 7 channels, while the multi-channel model uses only 2 out of the 7 channels

2.2. Initialization and pre-training

In this work we initialize the LSTM layers with parameters trained on 1 million hours of beamformed LFBE input [16]. As mentioned before, the main focus of this work is learning the front-end layers of the multi-channel acoustic model and the concomitant initialization. To this end, we examine various initialization strategies for the front-end layers.

In our baseline model, the spatial filtering component is initialized with a set of 12 super-directive beamformer coefficients that use the spherically isotropic noise field [17, 6], each with a different look direction, as described in [10]. This is the 'DSP-init' baseline. The output of the spatial filtering layer is then combined using an affine transform that is initialized using Xavier-normal [18], and the MFB affine transform with MFB weights. For reference, we also try initializing all three layers with Xavier-normal (cf.'Random-init').

We compare the standard initializations (DSP-init and Random-init) to data-driven pre-training of the front-end and joint training of the front-end and classification layers via T/S training. Our front end pre-training method is inspired by a minimum mean squared error beamformer approach [19]. We leverage the fact that we can compute the beamformed LFBE directly from 7 channel audio by using the super-directive beamformer [1]. With the LFBE features computed from beamformed audio, we pre-train the front-end layer with 2-channel raw input using L2-loss against the beam-

[1]For all the experiments, the beamformed audio and multi-channel audio were delay compensated.

Table 1. List of experiments, including the initialization used for the beamforming and MFB layers and the linear layer in-between, either random or based on a DSP technique. Each experiment is trained first using cross-entropy (Xent), then sMBR.

Experiment settings			
Model names	**Training dataset**	**Initialization technique**	**training technique**
Random-init-Xent + sMBR	621 hours	Xavier	Supervised
DSP-init-Xent + sMBR	621 hours	DSP, Xavier	Supervised
Pretrained-Xent + sMBR	pretrain: 1818, train: 621	Xavier + pre-training	Supervised
Pretrained-DSP-init-Xent + sMBR	pretrain: 1818, train: 621	DSP, Xavier + pre-training	Supervised
Student-DSP-init-Xent + sMBR	1818 hours	"DSP-init-sMBR" weights	Teacher-Student
Student-pretrained-Xent + sMBR	1818 hours	"Pretrained-sMBR" weights	Teacher-Student
Teacher sMBR	71500 hours	Uniform	Supervised

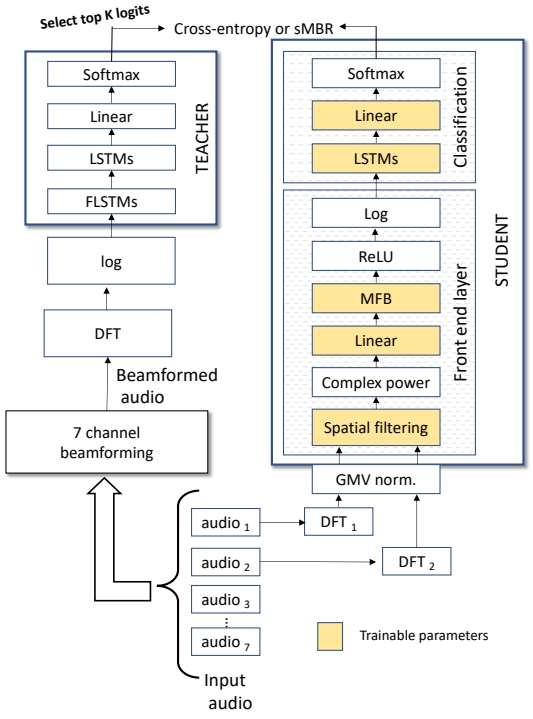

Fig. 2. Teacher-student training of the multi-channel acoustic model architecture. The left side shows the teacher architecture, and the right side shows the architecture of the model used both as a standalone and as a student model.

formed LFBE. This is shown in Figure 1. For pre-training the front-end with L2-loss, we experimented with initializing the model using DSP-based and random. When using DSP-based parameters, we initialize the linear layer in-between using a random uniform distribution whose minimum and maximum are the average of the beamformer and MFB minimum and maximum values. During pre-training, we fix the DSP-based beamforming and MFB weights for the first epoch, then fine-tune them along with the other parameters. Both setups converged best with batch size 16, learning rate 0.0001, and an Adam optimizer with $\beta = (0.9, 0.999)$ and $eps = 1e-08$. Interestingly, we found that the randomly initialized model converged faster and to a lower L2-loss than the model initialized with DSP-based parameters.

2.3. Teacher-student training

Our teacher model trained in a supervised manner on beamformed data. T/S training is semi-supervised, as it leverages untranscribed utterances fed to the teacher in the beamformed format and to the student in MC format. Additionally, the teacher model is larger and more complex than the student whose structure is designed for low latency, which makes this instance of T/S training knowledge distillation. Soft labels should be more informative than hard labels, providing the student the ability to learn more complex functions [13]. We use the techniques described in prior work on T/S training described in Section 1 [14, 15, 13]. Instead of using all 3183 senone probabilities, we only use the top 20 probabilities as described in [15]. This improves learning and saves space. We also soften the senone logits output by the teacher using temperature T. For all our experiments, we use $T = 2$ since that was found to be the optimal parameter in [15].

Our teacher model is trained using beamformed 256 dimensional log-DFT features. It is then used to output logits used for teacher-student training. The teacher model architecture consists of a 2-layer bi-directional F-LSTM [20] with hidden dimension 16, frequency dimension window 48, and stride dimension 15, followed by a 5-layer bidirectional LSTM with hidden dimension 768. The total number of weights in the model is 75.9M.

3. EXPERIMENT SETUP AND RESULTS

All of the data for these experiments was collected using a 7-channel circular microphone array described in [11]. The beamformed data, used for training the teacher model and generating the LFBE targets for the L2-loss based pre-initialization, was generated using the superdirective beamformer using the 7 channels as described in [10]. For all the multi-channel experiments, we selected 2 microphone channels from the opposite sides of the microphone array. We used

Index	Model name	WER Reduction %
1	**DSP-init-Xent + sMBR training** (baseline)	-
2	+ T/S training	27.3
3	Random-init-Xent + sMBR training	-65.3
4	Pretrained-DSP-init-Xent + sMBR training	-5.0
5	Pretrained-Xent + sMBR training	10.7
6	+ T/S training	31.0
7	**Teacher-biLSTM** (offline)	36.0

Table 2. Results by different initialization schemes and T/S training

621 hours of transcribed multi-channel data for supervised training. For T/S training and pre-training, we used 1818 hours of pooled transcribed and untranscribed data, which included the 621 hours. Even when the data was transcribed, we used the soft targets, not the transcriptions. The teacher model was trained on 71500 hours of transcribed beamformed data. The test set contains real-world far-field data, with a total of 58183 utterances for analysis. The MC AM was first trained in a supervised manner using cross entropy (Xent) and State-level Minimum Bayes Risk (sMBR) training, then used to initialize the student. For sMBR training of the student model, we just used the 621 hours of supervised data.

3.1. Results with different initialization techniques

A summary of our experiments is displayed in Table 1. For supervised training, we compare four different initialization techniques described in Section 2. The first two are initialized directly, either with DSP-based components (DSP-init, 1 in Table 2) or randomly (Random-init, 3 in Table 2). The next two add pre-training to the first two (Pretrained-DSP-init and Pretrained, 4 and 5 respectively in Table 2). The results are displayed in Table 2. We use the "DSP-init" model (1 in Table 2) as our baseline because it produces better results than random initialization of the front-end. Pre-training on randomly initialized linear transformations showed a relative WER improvement of 10.7% compared to the baseline. This result demonstrates that the front-end can be learned in a fully data-driven manner.

We observe that the pre-training works better when initialized with random weights rather than with DSP weights, we hypothesize that this is because the conventional beamformer weights are sub-optimal for the real-data conditions and initializing the parameters with those weights prevents the model from getting to the optimal parameter set with limited data.

3.2. Results with T/S training

We select the two best performing models (DSP-init and Pretrained) to initialize the student (Student-DSP-init and Student-pretrained) for T/S training and the results are showed in Table 2. T/S training improved the WER by 27.3% relative compared to the "DSP-init" baseline and helped the model perform nearly as well as the teacher despite being smaller (2 in Table 2). Combining pre-training and T/S training improves the the WER by 31.0% relative (6 in Table 2). For all our experiments with initialization, we saw 13-15% relative WER improvements with sMBR training on top of cross entropy training. However with the T/S training the improvements were 3-5% relative, likely because it only uses transcribed data.

The gains we observe using T/S training are in the same range as the results reported in [15], and our experiments demonstrate that we can distill information from a single channel teacher model to a multi-channel student model and learn the front-end components in a data-driven manner. The second observation we make is the WER improvements with pre-training improve performance even after T/S training. This result shows us that we do not need to rely on prior knowledge in order to train these models, except for the target LFBE values from beamformed audio required to pre-train the front end weights.

4. CONCLUSIONS

In this paper, we explore using semi-supervised learning for optimizing the fully learnable front-end of a multi-channel acoustic model The teacher-student approach adopted in our experiments leverages untranscribed data and distills knowledge from a complex "teacher" model that uses beamformed data to a low-latency multi-channel "student" model. We also introduce pre-training of the front-end, learning the beamforming and feature extraction layers with the beamformed LFBEs as the target. The target is computed directly from audio, making it once again possible to harness untranscribed data. We find that both techniques improve the performance of our model.

In this work, we have only studied these techniques applied to two channels input data. In the future, we would like to apply the same techniques to a large number of multi-channel inputs. We would also like to explore knowledge distillation for state-minimum Bayes risk training [21] to see if we can get more gains from the sequence training stage. Finally we would also like to explore other architectures for multi-channel acoustic modeling.

5. REFERENCES

[1] M. Wölfel and J. W. McDonough, *Distant speech recognition*, Wiley Online Library, 2009.

[2] M. Delcroix, T. Yoshioka, A. Ogawa, Y. Kubo, M. Fujimoto, N. Ito, K. Kinoshita, M. Espi, T. Hori, T. Nakatani, et al., "Linear prediction-based dereverberation with advanced speech enhancement and recognition technologies for the REVERB challenge," in *Reverb workshop*, 2014.

[3] J. Benesty, J. Chen, and Y. Huang, *Microphone array signal processing*, vol. 1, Springer Science & Business Media, 2008.

[4] M. Brandstein and D. Ward, *Microphone arrays: signal processing techniques and applications*, Springer Science & Business Media, 2013.

[5] K. Kumatani, T. Arakawa, K. Yamamoto, J. McDonough, B. Raj, R. Singh, and I. Tashev, "Microphone array processing for distant speech recognition: Towards real-world deployment," in *Proc. Asia Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2012, pp. 1–10.

[6] Ivan Himawan, Iain McCowan, and Sridha Sridharan, "Clustered blind beamforming from ad-hoc microphone arrays," *Transactions Audio, Speech, and Language Processing (TASLP)*, vol. 19, no. 4, pp. 661–676, 2010.

[7] T. M. Sullivan and R. M. Stern, "Multi-microphone correlation-based processing for robust speech recognition," in *Int. Conf. Acoustics, Speech, and Signal Processing (ICASSP)*. IEEE, 1993, vol. 2, pp. 91–94.

[8] T. N. Sainath, R. J. Weiss, K. W. Wilson, B. Li, A. Narayanan, E. Variani, M. Bacchiani, I. Shafran, A. Senior, K. Chin, et al., "Multichannel signal processing with deep neural networks for automatic speech recognition," *Transactions Audio, Speech, and Language Processing (TASLP)*, vol. 25, no. 5, pp. 965–979, 2017.

[9] T. Ochiai, S. Watanabe, T. Hori, and J. R. Hershey, "Multichannel end-to-end speech recognition," in *Proc. 34th Int. Conf. Machine Learning (ICML)*. JMLR, 2017, vol. 70, pp. 2632–2641.

[10] M. Wu, K. Kumatani, S. Sundaram, N. Ström, and B. Hoffmeister, "Frequency domain multi-channel acoustic modeling for distant speech recognition," in *Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6640–6644.

[11] K. Kumatani, M. Wu, S. Sundaram, N. Ström, and B. Hoffmeister, "Multi-geometry spatial acoustic modeling for distant speech recognition," in *Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6635–6639.

[12] Jinyu Li, Rui Zhao, Jui-Ting Huang, and Yifan Gong, "Learning small-size dnn with output-distribution-based criteria," in *Fifteenth annual conference of the international speech communication association*, 2014.

[13] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *stat*, vol. 1050, pp. 9, 2015.

[14] J. Li, M. L. Seltzer, X. Wang, R. Zhao, and Y. Gong, "Large-scale domain adaptation via teacher-student learning," *Proc. Interspeech*, 2017.

[15] L. Mošner, M. Wu, A. Raju, S. H. K. Parthasarathi, K. Kumatani, S. Sundaram, R. Maas, and B. Hoffmeister, "Improving noise robustness of automatic speech recognition via parallel data and teacher-student learning," in *Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6475–6479.

[16] S. H. K. Parthasarathi and N. Strom, "Lessons from building acoustic models with a million hours of speech," in *Int. Conf. Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 6670–6674.

[17] S. Doclo and M. Moonen, "Superdirective beamforming robust against microphone mismatch," *Transactions Audio, Speech, and Language Processing (TASLP)*, vol. 15, no. 2, pp. 617–631, 2007.

[18] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. 13th Int. Conf. Artificial Intelligence and Statistics (AISTATS)*, 2010, pp. 249–256.

[19] Yonina C Eldar, Arye Nehorai, and Patricio S La Rosa, "A competitive mean-squared error approach to beamforming," *IEEE Transactions on Signal Processing*, vol. 55, no. 11, pp. 5143–5154, 2007.

[20] Ji. Li, A. Mohamed, G. Zweig, and Y. Gong, "Lstm time and frequency recurrence for automatic speech recognition," in *Workshop Automatic Speech Recognition and Understanding (ASRU)*. IEEE, 2015, pp. 187–191.

[21] N. Kanda, Y. Fujita, and K. Nagamatsu, "Investigation of lattice-free maximum mutual information-based acoustic models with sequence-level kullback-leibler divergence," in *Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2017, pp. 69–76.