INTERSPEECH 2021
30 August – 3 September, 2021, Brno, Czechia

Investigation of Practical Aspects of Single Channel Speech Separation for ASR

Jian Wu[1], *Zhuo Chen*[2], *Sanyuan Chen*[1], *Yu Wu*[1], *Takuya Yoshioka*[2], *Naoyuki Kanda*[2], *Shujie Liu*[1], *Jinyu Li*[2]

[1]Microsoft, China
[2]Microsoft, USA

{wujian, zhuc, v-sanych, yuwu1, tayoshio, nakanda, shujliu, jinyli}@microsoft.com

Abstract

Speech separation has been successfully applied as a front-end processing module of conversation transcription systems thanks to its ability to handle overlapped speech and its flexibility to combine with downstream tasks such as automatic speech recognition (ASR). However, a speech separation model often introduces target speech distortion, resulting in a sub-optimum word error rate (WER). In this paper, we describe our efforts to improve the performance of a single channel speech separation system. Specifically, we investigate a two-stage training scheme that firstly applies a feature level optimization criterion for pre-training, followed by an ASR-oriented optimization criterion using an end-to-end (E2E) speech recognition model. Meanwhile, to keep the model light-weight, we introduce a modified teacher-student learning technique for model compression. By combining those approaches, we achieve a absolute average WER improvement of 2.70% and 0.77% using models with less than 10M parameters compared with the previous state-of-the-art results on the LibriCSS dataset for utterance-wise evaluation and continuous evaluation, respectively.

Index Terms: speech separation, speech recognition, single channel, joint training, teacher student learning

1. Introduction

Deep learning based speech separation has been investigated in recent years since the proposal of deep clustering (DPCL) [1] and permutation invariant training (PIT) [2]. Various follow-up studies have been reported, including exploration of the different architectures [3, 4, 5] and recipes [6, 7, 8], extension to multi-channel processing [9, 10] and joint modeling with other tasks such as automatic speech recognition (ASR) [11, 12]. These advances resulted in improved performance with respect to several metrics related to audio quality and ASR accuracy.

Recently, the multi-channel speech separation achieves good performance [13, 14] and has been successfully integrated into conversation transcription systems [15]. However, the improvement has still been limited with single channel input for the conversational tasks [16, 17, 18]. The single channel conversation transcription remains challenging for two reasons. Firstly, a single channel network can not be benefited from the spatial information, leading to inferior separation. Secondly, the single channel separation often employs a signal-level objective function, which is known to introduce speech distortion that hurts the accuracy of ASR systems [19].

To overcome this limitation, several methods have been studied. In [20, 21], a feature-level objective function was found to be beneficial for ASR. The joint optimization of the front-end with the back-end ASR was also shown to be effective in both speech enhancement [22, 23, 24] and separation [25, 26]. However, despite these promising results, the past studies on ASR-oriented speech separation lacks consideration of two aspects. Firstly, the ASR back-end used in the past works were either jointly updated or tightly connected to the separation model, which could cause undesirable word error rate (WER) degradation on out-of-domain dataset and less flexibility of using each module in different application scenarios. Secondly, a jointly trained speech separation model may suffer from performance degradation when the back-end ASR model is changed.

In this paper, we attempt to answer the above questions experimentally by using the LibriCSS dataset with the goal of improving the performance of the single channel speech separation for conversation transcription. We adopt a training recipe which can leverage knowledge from both an ASR system and spectrum reconstruction. Specifically, our training recipe contains two stages, where a seed model is firstly trained under a conventional mask-based feature approximation objective, followed by fine-tuning with the ASR-oriented training criteria using an end-to-end ASR network. The ASR model used for training could be different from that employed for evaluation, and its parameters are kept fixed during training to prevent the co-adaptation of the front-end and ASR model. In addition, our comparison shows that a Conformer-based separation model outperforms several other state-of-the-art model architectures. As regards the model architecture, we also examine the effect of model compression using the layer-wise teacher-student learning proposed recently [27] to achieve fast inference and lower runtime cost. Finally, we report results for both the utterance-wise and continuous evaluation settings of LibriCSS, outperforming the previously reported best numbers.

2. System Description

2.1. Signal Model

We consider the following signal model with C speakers. The single channel far-field signal $\mathbf{y}$ in time domain is impaired by reverberation and additive noise:

$$\mathbf{y} = \sum_c \mathbf{x}_c + \mathbf{n} = \sum_c \mathbf{s}_c * \mathbf{h}_c + \mathbf{n}, \tag{1}$$

where $\mathbf{h}_c$ is room impulse response (RIR) between speaker c ($0 \leqslant c < C$) and the microphone. $\mathbf{s}_c$ is c-th source speaker and $\mathbf{x}_c$ denotes the corresponding image signal. We model environment noise as $\mathbf{n}$, consisting of directional and isotropic noise. After applying short-time Fourier transform (STFT), (1) is converted into frequency domain:

$$\mathbf{Y} = \sum_c \mathbf{X}_c + \mathbf{N}, \tag{2}$$

http://dx.doi.org/10.21437/Interspeech.2021-921

where $\{\mathbf{Y}, \mathbf{N}, \mathbf{X}_c\} \in \mathbb{C}^{T\times F}$. T and F denote the total number of time frames and frequency bins.

2.2. Conformer Structure

We employ the frequency domain model $\mathcal{M}(\cdot)$ for single channel separation as it was found to be more robust for speech recognition than time domain structures from our preliminary results. The network is trained to estimate the time-frequency masks (TF-masks) on image signals, i.e., $\mathbf{M}_{0,1} = \mathcal{M}(\mathbf{Y}) \in \mathbb{R}^{T\times F}$. Inverse STFT (iSTFT) is used for signal reconstruction for the additional post processing

$$\mathbf{x}'_c = \text{iSTFT}(\mathbf{Y} \odot \mathbf{M}_c). \tag{3}$$

Following the work in [18], a modified Conformer [28] structure that consists a stack of conformer blocks is used as $\mathcal{M}(\cdot)$ for TF-masks estimation. Our conformer block is composed of three parts, i.e., a multi-head self-attention module (MHSA) [29], a convolution module (CONV) and a feedforward network (FFN). The output of the conformer block z_3 is calculated as the following equations given the input z_0

$$\begin{aligned} z_1 &= z_0 + \text{MHSA}(\ln(z_0)) \\ z_2 &= z_1 + \text{CONV}(\ln(z_1)) \\ z_3 &= z_2 + \text{FFN}(\ln(z_2)) \end{aligned} \tag{4}$$

where $\ln(\cdot)$ denotes the layer normalization. The dropout layers within the conformer block are disabled.

For the implementation of the MHSA, we adopt the learnt relative position encoding described in [30] instead of the original version [31]. Besides, a squeeze-and-excitation (SE) [32] block is added to the last layer of the convolution module which operates on the output of the second pointwise convolution.

2.3. Network Training

Although the prior work in [18] has reported significant improvement in terms of WER by using the Conformer structure for speech separation, a large performance gap is observed between the single channel and multi-channel systems, i.e., 5.9% and 8.2% absolute WER increase when overlap ratio is 30% and 40%. One of the potential reasons lies in the mismatch between the front-end objective function and speech recognition. In a typical mask learning based separation network, ideal amplitude mask (IAM) is used as the training target and the loss function for spectrum approximation is defined as

$$\mathcal{L}_{\text{SA}} = \arg\min_{\phi\in\mathcal{P}} \sum_{(i,j)\in\phi} \|\mathbf{M}_i \odot |\mathbf{Y}| - |\mathbf{X}_j|\|_F \tag{5}$$

under the permutation invariant training criteria. $\mathcal{P}$ refers all the possible permutations over $C = 2$ speakers and $\|\cdot\|_F$ denotes the Frobenius norm.

However, the modern ASR model uses the acoustic features such as filter-bank (fbank), MFCC as the network input, which employs a representation with considerably lower dimension than acoustic mask. Thus the spectrum approximation in front-end may be redundant for the reconstruction of the acoustic features, which not only increases the difficulties of the network optimization, but also puts mismatched emphasis for signal recovery. For example, the fbank feature gives more weights on the lower frequency regions, while the IAM treats each frequency equally.

To lead the better estimation of the acoustic features that the ASR back-end expected and reduce the potential feature distortion, we modified equation (5) to measure the feature level difference instead of the amplitude:

$$\mathcal{L}_{\text{FA}} = \arg\min_{\phi\in\mathcal{P}} \sum_{(i,j)\in\phi} \|\mathcal{F}(\mathbf{M}_i \odot |\mathbf{Y}|) - \mathcal{F}(|\mathbf{X}_j|)\|_F \tag{6}$$

where $\mathcal{F}(\cdot)$ denotes the feature transform function. In order to further match the ASR input, on top of the model optimized with (6), we adopt a well trained ASR model and tune the separation model using ASR's training criteria. In this work, during training, we employ encoder-decoder based E2E speech recognition model in the experiments, with a hybrid CTC & attention objective function [33]:

$$\mathcal{L}_{\text{ASR}} = \sum_c \lambda \log p_{\text{ctc}}(\mathbf{r}_c|\mathbf{X}'_c) + (1-\lambda)\log p_{\text{dec}}(\mathbf{r}_c|\mathbf{X}'_c) \tag{7}$$

where $\mathbf{X}'_c = \mathcal{F}(\mathbf{M}_c \odot |\mathbf{Y}|)$ denotes the feature sequence of the separated speaker c and $\mathbf{r}_c$ is the corresponding transcription. λ is a hyper parameter to balance the CTC loss on encoder branch $\log p_{\text{ctc}}(\cdot)$ and the cross-entropy loss $\log p_{\text{dec}}(\cdot)$ on the decoder predictions.

When initialized with a well trained separation model using feature recovery objective (6), the permutation computation logic is not necessary for ASR objective, as shown in (7). For each training sample, we determine the label permutation by measuring the distance between $\mathbf{M}_c \odot |\mathbf{Y}|$ and spectrum of the reference signal $|\mathbf{X}_c|, c \in \{0, 1\}$. In this case we can save 50% computation when calculating the loss values. Note that we found that directly optimizing with ASR objective from scratch leads to strong turbulence in model convergence and results in inferior performance, so we excluded this setup in the experiments.

2.4. Model Compression

Despite the promising results, the conformer based separation network usually employs model architecture with large parameter size. This usually brings difficulties in model deployment because of expensive computation and slow inference, especially on edge devices. To reduce the model size, we apply the recently advanced teacher student learning. Specifically, the well trained large separation network serves as the teacher and provides the reference separation for the smaller student model. The layer-wise TS objective $\mathcal{L}_{\text{LTS}}$ adds L2 distance between the hidden representation from teacher and student model in each layer, as shown in (8):

$$\mathcal{L}_{\text{LTS}} = \gamma_S \cdot \mathcal{L}_{\text{TS}} + \sum_s \gamma_s \cdot \left\|\mathbf{H}_s - \mathbf{H}^{\top}_{g(s)}\right\|_F, \tag{8}$$

where $g(\cdot)$ is a uniform layer mapping function between indices from student layers to teacher layers and γ_s is the weight value of the s-th hidden layer. $\mathcal{L}_{\text{TS}}$ removes the permutation computation to force the student network fully following the teacher's actions:

$$\mathcal{L}_{\text{TS}} = \sum_c \left\|\mathcal{F}(\mathbf{M}_c \odot |\mathbf{Y}|) - \mathcal{F}(\mathbf{M}^{\top}_c \odot |\mathbf{Y}|)\right\|_F, \tag{9}$$

where the $\mathbf{M}^{\top}_c$ refers to teacher's mask prediction of the speaker c.

Moreover, we apply an *objective shifting* (OS) [34] mechanism for more effective TS learning, which enables us to train the student network with both teacher's predictions and ground truth references. The objective function equipped with OS mechanism is:

$$\mathcal{L}_{\text{OS}} = \omega_t \mathcal{L}_{\text{FA}} + (1 - \omega_t)\mathcal{L}_{\text{LTS}}, \tag{10}$$

where $\omega_t = \text{sigmoid}(-k(t - t_0))$ and t refers to the training steps. k and t_0 are hyper parameters. We describe the detailed exploration on separation with TS learning in a separate paper [35], and we refer audiences to that for more details.

3. Experimental Setup

3.1. Dataset

We evaluate our methods on the LibriCSS [16] dataset which consists of 10 hours of multi-speaker recording from a meeting room with an overlap ratio ranging from 0% to 40%. The recording device is a seven-channel circular microphone array and we use the signal of the first channel for the single channel performance evaluation. We evaluate our model in both utterance-wise and continuous evaluation as defined in [16].

For model training, we employ two datasets: the first one consists of 219-hour data using the close talk speech sampled from WSJ1 and the second one includes 1000-hour overlapped mixture whose source speech is sampled from LibriSpeech. Both dataset are simulated according to the signal model described in equation (1) and each training sample contains one or two speakers. The artificial room impulse responses are generated using the image method. For two-speaker cases, the mixing SDR ranges from -5 dB to 5 dB and four mixture types are considered following the work [36]. Both directional and isotropic noises were added to each mixture with an SNR uniformly sampled between [0, 20] dB and [10, 20] dB, respectively. The directional noises are simulated by convolving the point source noise from MUSAN dataset with the RIRs.

3.2. Separation Model

Following the description in Section 2, the Conformer structure is adopted as our primary network. The $\text{Cfmr}_{\text{base}}$ model has 16 encoder layers, 256 attention dimensions with 4 heads. The inner-layer of the FFN has 1024 dimensions and the kernel size and channel number used in CONV is 33 and 512, respectively. The $\text{Cfmr}_{\text{small}}$ model has the similar configurations but reduces the layer of the encoders to 6.

We also evaluate a Transformer, a Convolution Recurrent Network (CRN) and a Dense-CRN network as representative baseline systems. The Transformer model used here includes 16 encoder layers with the 4 head, 256 dimensional MHSA (using relative position encoding) and 1024 dimensional FFN. The CRN structure is a real-valued version of the DCCRN [37], and consists of a 7-layer encoder/decoder with 3 layer bidirectional LSTMs. For Dense-CRN, we insert the DenseNet block between the layers in CRN's encoder and decoder, similar to the structure used in [17]. The Dense-CRN consists of a 8-layer encoder/decoder with 2 layer BLSTMs.

The 25 ms frame size with the frame shift of 10 ms is used for feature generation. A 512-point FFT size and hamming window are used in (i)STFT, forming the 257-dimentional masks and spectrum. The log spectrogram with utterance-wise mean variance normalization is extracted as the input feature for all the separation models. We only consider mixture with at most two speakers in experiments. The sigmoid$(\cdot)$ function is applied to final layer to make sure the masks have the value between 0 and 1.

3.3. ASR Model

We report our single channel results on two ASR back-end models. Both of them are trained on 960 hours of LibriSpeech training data, using the word piece units of the transcription as target. The first ASR, named $\text{ASR}_{\text{matched}}$, is the ASR model used for ASR-oriented training of speech separation following (7). It consists of 12 Conformer [18] encoder layers and 6 Transformer decoder layers with 80-dimentional log fbank as the input feature. This model shows WERs of 2.80% and 6.80% on Librispeech *test-clean* and *test-other*, respectively. The second ASR is the one developed in [38], which we call ASR [38]. A concatenation of the filter bank and pitch features are used as the input to this model, and it achieves WERs of 2.08% and 4.95% on *test-clean* and *test-other*, respectively. For both ASR systems, an external language model trained on LibriSpeech's text corpus is applied with shallow fusion. The beam size and other decoding hyper-parameters are tuned on LibriSpeech *dev-other* set.

3.4. Training Details

In our experiments, the training schemes are different depends on the model initialization. When the model is randomly initialized, the Transformer, $\text{Cfmr}_{\text{base}}$ and $\text{Cfmr}_{\text{small}}$ models are trained by $\mathcal{L}_{\text{FA}}$ with AdamW optimizer where the weight decay is set to 0.01. A learning rate scheduler with linear warm-up and decay is used and the peak value of the learning rate is set to 10^{-4}. The model is trained for 260k steps in total where the warm-up step is set as 10k. The CRN and Dense-CRN models are trained with the Adam optimizer with a weight decay of 10^{-5} and an initial learning rate of 10^{-3}. Those networks are trained for a maximum of 260k steps and the learning rate is halved if no validation improvement is observed for two consecutive epochs. The early stopping strategy is applied to avoid over-fitting.

When using the ASR-oriented objective function $\mathcal{L}_{\text{ASR}}$ or retraining the model by $\mathcal{L}_{\text{FA}}$ with additional data, the network is initialized with the pre-trained models, and we continue the training for 100k steps. During the first 25k steps, the learning rate is set to 4×10^{-5} and starts linear decay after that. The AdamW optimizer with a weight decay of 0.01 is used. For $\mathcal{L}_{\text{ASR}}$, λ is set to 0.2 and the unigram label smoothing is involved for calculation of the cross-entropy loss, which are kept same as the training configuration of $\text{ASR}_{\text{matched}}$. We apply gradient accumulation of 4 steps to increase the batch size due to the memory issues. The linear mel transform is used as the function $\mathcal{F}(\cdot)$ in our experiments.

For TS learning, the training configuration is same with the random initialization training, e.g., the learning rate scheduler, optimizer and the total training steps. We use $t = 1.5 \times 10^5$ and $k = 5 \times 10^{-4}$ in $\mathcal{L}_{\text{OS}}$ and $g(s) = \{2, 5, 8, 11, 14, 15\}$ for $s \in \{0, \cdots, 5\}$ in $\mathcal{L}_{\text{LTS}}$. All the models are trained using our self-developing tools based on PyTorch.

4. Evaluation Results

4.1. Architecture Comparison

We compare the utterance-wise performance on different network architectures in Table 1. Here, all front-end models are trained with 219 hours WSJ1 dataset, and $\text{ASR}_{\text{matched}}$ is used

Table 1: *WER (%) for utterance-wise evaluation with different separation model architectures. $ASR_{matched}$ is used.*

Separation	#Param	Loss	Overlap Ratio (%)					
			0S	0L	10	20	30	40
No separation	-	-	6.0	5.5	12.2	20.3	28.7	38.3
$Cfmr_{base}$	26.03M	$\mathcal{L}_{SA}$	**5.7**	**5.3**	7.7	10.7	13.4	15.1
		$\mathcal{L}_{FA}$	5.8	5.4	**7.4**	**10.0**	**12.1**	**13.9**
$Cfmr_{small}$	9.97M	$\mathcal{L}_{FA}$	6.0	5.3	8.1	11.0	13.4	15.3
Transformer	12.97M		5.7	5.4	8.3	12.0	15.0	17.4
CRN	20.37M		6.1	5.8	8.4	12.5	16.5	19.3
Dense-CRN	17.99M		5.9	5.7	8.2	11.6	14.6	17.3

Table 2: *WER (%) for utterance-wise evaluation with different training objective function and ASR back-end.*

ASR	Separation	Loss	Overlap Ratio (%)					
			0S	0L	10	20	30	40
$ASR_{matched}$	$Cfmr_{base}$	$\mathcal{L}_{FA}$	5.4	5.0	6.8	8.9	10.4	11.7
		$\mathcal{L}_{ASR}$	5.1	4.7	6.4	8.5	9.8	10.7
		$\mathcal{L}_{FA} \rightarrow \mathcal{L}_{ASR}$	**4.8**	**4.6**	**6.1**	**8.0**	**9.2**	**10.0**
ASR[38]	$Cfmr_{base}$	$\mathcal{L}_{FA}$	3.7	3.8	5.0	6.9	8.6	9.8
		$\mathcal{L}_{ASR}$	3.7	3.7	5.1	6.4	8.1	9.2
		$\mathcal{L}_{FA} \rightarrow \mathcal{L}_{ASR}$	**3.6**	**3.6**	**4.7**	**6.4**	**7.7**	**8.4**
	[18]	$\mathcal{L}_{SA}$	5.4	5.0	7.5	10.7	13.8	17.1
[17]	$SISO_{1+3}$	[17]	4.9	5.1	6.7	9.4	12.7	15.5

for the back-end. On Conformer architecture, a clear WER improvement is observed when comparing the feature-level and the signal-level objectives, especially for high overlapped conditions. For example, WER on $30\%/40\%$ subsets are reduced from $13.4\%/15.1\%$ to $12.1\%/13.9\%$. We also observe that both $Cfmr_{base}$ and $Cfmr_{small}$ significantly outperform other popular network architectures. Compared with $Cfmr_{base}$, smaller model $Cfmr_{small}$ showed a clear WER increase, indicating that the necessity of more efficient training scheme such as model compression.

4.2. Comparison of Training Objective

The comparison among models trained with different objective functions is shown in Table 2. Here, all models are trained on the 1000-hour LibriSpeech mixture data starting from the pre-trained model with 219-hour WSJ1 mixture. The first and fourth rows indicate the results of the speech separation model retrained by $\mathcal{L}_{FA}$ while the second and fifth rows indicate the results of the model retrained by $\mathcal{L}_{ASR}$. Finally, the rows with a loss $\mathcal{L}_{FA} \rightarrow \mathcal{L}_{ASR}$ refers the model using ASR-oriented objective, but initialized with the model corresponding to the first row.

Compared with Table 1, a clear benefit of using additional training data can be observed. The model retrained by $\mathcal{L}_{FA}$ reduces the average WER from 9.1% to 8.0% on $ASR_{matched}$ and the model trained by $\mathcal{L}_{ASR}$ demonstrates advantageous performance than $\mathcal{L}_{FA}$ on all subsets, showing the effectiveness of ASR-oriented optimization. Meanwhile, another 5% WER reduction is observed by better initialization ($\mathcal{L}_{FA} \rightarrow \mathcal{L}_{ASR}$). We can also see that, even though the separation model is tuned with $ASR_{matched}$, the consistent performance improvement still exists when the ASR model is changed to ASR [38] which uses a slightly different feature and architecture. Finally, compared to the state-of-the-art system reported in [17], our model shows a significant performance improvement, reducing the average WER from 9.1% to 5.7%.

Table 3: *WER (%) for utterance-wise evaluation with $Cfmr_{small}$ trained by the TS learning and ASR-oriented objective function.*

ASR	Loss	Overlap Ratio (%)					
		0S	0L	10	20	30	40
ASR[38]	$\mathcal{L}_{FA}$	3.8	3.7	5.5	7.5	10.1	11.7
	$\mathcal{L}_{OS}$	3.8	3.7	**5.3**	7.1	9.5	10.7
	$\mathcal{L}_{ASR}$	3.9	3.9	5.7	7.7	9.4	10.5
	$\mathcal{L}_{OS} \rightarrow \mathcal{L}_{ASR}$	**3.7**	**3.8**	5.4	**7.0**	**8.7**	**9.5**

Table 4: *WER (%) for continuous evaluation. ASR[38] is used.*

Separation	Loss	Overlap Ratio (%)					
		0S	0L	10	20	30	40
[18]	$\mathcal{L}_{SA}$	**6.9**	**6.1**	9.1	12.5	16.7	19.3
$Cfmr_{base}$	$\mathcal{L}_{ASR}$	7.3	6.2	8.9	9.9	13.1	13.9
	$\mathcal{L}_{FA} \rightarrow \mathcal{L}_{ASR}$	7.5	6.4	**8.4**	**9.4**	**12.4**	**13.2**
$Cfmr_{small}$	$\mathcal{L}_{ASR}$	11.6	10.3	13.0	13.7	17.3	17.8
	$\mathcal{L}_{OS} \rightarrow \mathcal{L}_{ASR}$	8.2	6.5	10.0	11.2	14.5	15.6

4.3. Model Compression

The result for model comparison is shown in Table 3, where all models have $Cfmr_{small}$ architecture. $\mathcal{L}_{OS} \rightarrow \mathcal{L}_{ASR}$ means the model trained using ASR-oriented objective, with $\mathcal{L}_{OS}$ optimized model as the initialization. All the models are trained on 1000-hour Librispeech dataset.

Similar to Table 2, the two-stage training scheme significantly improves the WER of $Cfmr_{small}$. Compared with a training from scratch, $\mathcal{L}_{OS}$ brings notable WER reduction and shows competitive results with the ASR-oriented training. A further WER reduction is observed when the $\mathcal{L}_{OS}$ trained model is further fine tuned by the ASR-oriented objective function. After applying the TS learning, the performance gap between teacher and student models is reduced to 1.1% WER increase, with 62% reduction in parameter size.

4.4. Continuous Evaluation

We follow the similar chunk-wise processing described in [18] for continuous speech separation with signal-wise stitching. Compared with the previous work in [18], both $Cfmr_{base}$ and $Cfmr_{small}$ show remarkable improvement for the highly overlapping test sets after the training based on $\mathcal{L}_{ASR}$, while the $Cfmr_{small}$ has a 5 times smaller parameter size compared with the 58.72M-parameter model of [18]. However, the numbers on the non-overlapping sets has slight performance degradation compared with the results of [16], which suggests us the potential room of improvement under current framework.

5. Conclusions

In this paper, we investigate several practical aspects of single channel speech separation for ASR, including a two-stage training scheme to utilize both feature and ASR-oriented optimization criterion and the TS training as the final step to compress the model size. The conclusions on the utterance-wise and continuous evaluations are consistent and the performance gains from the ASR-oriented training could be shifted to another different ASR model. The state-of-the-art results using a smaller Conformer model with less than 10M parameters on LibriCSS dataset are achieved on utterance-wise evaluation, which gives an average 2.7% absolute WER reduction compared with the best results shown before. For the continuous evaluation, we achieve an average relative WER improvement of 6.4% with significant gains on overlapped sets.

6. References

[1] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *ICASSP*. IEEE, 2016, pp. 31–35.

[2] M. Kolbæk, D. Yu, Z.-H. Tan, and J. Jensen, "Multitalker speech separation with utterance-level permutation invariant training of deep recurrent neural networks," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 10, pp. 1901–1913, 2017.

[3] Y. Luo, Z. Chen, and N. Mesgarani, "Speaker-independent speech separation with deep attractor network," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 4, pp. 787–796, 2018.

[4] Z.-Q. Wang, J. Le Roux, and J. R. Hershey, "Alternative objective functions for deep clustering," in *ICASSP*. IEEE, 2018, pp. 686–690.

[5] Y. Luo and N. Mesgarani, "Conv-tasnet: Surpassing ideal time–frequency magnitude masking for speech separation," *IEEE/ACM transactions on audio, speech, and language processing*, vol. 27, no. 8, pp. 1256–1266, 2019.

[6] Z.-X. Li, Y. Song, L.-R. Dai, and I. McLoughlin, "Listening and grouping: an online autoregressive approach for monaural speech separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 4, pp. 692–703, 2019.

[7] N. Takahashi, S. Parthasaarathy, N. Goswami, and Y. Mitsufuji, "Recursive speech separation for unknown number of speakers," *arXiv preprint arXiv:1904.03065*, 2019.

[8] M. Ge, C. Xu, L. Wang, E. S. Chng, J. Dang, and H. Li, "Spex+: A complete time domain speaker extraction network," *arXiv preprint arXiv:2005.04686*, 2020.

[9] Y. Luo, Z. Chen, N. Mesgarani, and T. Yoshioka, "End-to-end microphone permutation and number invariant multi-channel speech separation," in *ICASSP*. IEEE, 2020, pp. 6394–6398.

[10] Z.-Q. Wang and D. Wang, "Combining spectral and spatial features for deep learning based blind speaker separation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 27, no. 2, pp. 457–468, 2018.

[11] S. Settle, J. Le Roux, T. Hori, S. Watanabe, and J. R. Hershey, "End-to-end multi-speaker speech recognition," in *ICASSP*. IEEE, 2018, pp. 4819–4823.

[12] X. Chang, Y. Qian, K. Yu, and S. Watanabe, "End-to-end monaural multi-speaker asr system without pretraining," in *ICASSP*. IEEE, 2019, pp. 6256–6260.

[13] N. Kanda, C. Boeddeker, J. Heitkaemper, Y. Fujita, S. Horiguchi, K. Nagamatsu, and R. Haeb-Umbach, "Guided source separation meets a strong asr backend: Hitachi/paderborn university joint investigation for dinner party asr," *arXiv preprint arXiv:1905.12230*, 2019.

[14] J. Wu, Z. Chen, J. Li, T. Yoshioka, Z. Tan, E. Lin, Y. Luo, and L. Xie, "An end-to-end architecture of online multi-channel speech separation," *arXiv preprint arXiv:2009.03141*, 2020.

[15] T. Yoshioka, I. Abramovski, C. Aksoylar, Z. Chen, M. David, D. Dimitriadis, Y. Gong, I. Gurvich, X. Huang, Y. Huang *et al.*, "Advances in online audio-visual meeting transcription," in *Proc. ASRU*, 2019, pp. 276–283.

[16] Z. Chen, T. Yoshioka, L. Lu, T. Zhou, Z. Meng, Y. Luo, J. Wu, X. Xiao, and J. Li, "Continuous speech separation: dataset and analysis," in *ICASSP*. IEEE, 2020, pp. 7284–7288.

[17] Z.-Q. Wang, P. Wang, and D. Wang, "Multi-microphone complex spectral mapping for utterance-wise and continuous speaker separation," *arXiv preprint arXiv:2010.01703*, 2020.

[18] S. Chen, Y. Wu, Z. Chen, J. Wu, J. Li, T. Yoshioka, C. Wang, S. Liu, and M. Zhou, "Continuous speech separation with conformer," *arXiv e-prints*, pp. arXiv–2008, 2020.

[19] S.-J. Chen, A. S. Subramanian, H. Xu, and S. Watanabe, "Building state-of-the-art distant speech recognition using the chime-4 challenge with a setup of speech enhancement baseline," *arXiv preprint arXiv:1803.10109*, 2018.

[20] P. Wang and D. Wang, "Enhanced spectral features for distortion-independent acoustic modeling." in *INTERSPEECH*, 2019, pp. 476–480.

[21] Q. Wang, I. L. Moreno, M. Saglam, K. Wilson, A. Chiao, R. Liu, Y. He, W. Li, J. Pelecanos, M. Nika *et al.*, "Voicefilter-lite: Streaming targeted voice separation for on-device speech recognition," *arXiv preprint arXiv:2009.04323*, 2020.

[22] 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 *ICASSP*. IEEE, 2015, pp. 4375–4379.

[23] Z.-Q. Wang and D. Wang, "A joint training framework for robust automatic speech recognition," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 24, no. 4, pp. 796–806, 2016.

[24] T. Menne, R. Schlüter, and H. Ney, "Investigation into joint optimization of single channel speech enhancement and acoustic modeling for robust asr," in *ICASSP*. IEEE, 2019, pp. 6660–6664.

[25] T. von Neumann, K. Kinoshita, L. Drude, C. Boeddeker, M. Delcroix, T. Nakatani, and R. Haeb-Umbach, "End-to-end training of time domain audio separation and recognition," in *ICASSP*. IEEE, 2020, pp. 7004–7008.

[26] T. von Neumann, C. Boeddeker, L. Drude, K. Kinoshita, M. Delcroix, T. Nakatani, and R. Haeb-Umbach, "Multi-talker asr for an unknown number of sources: Joint training of source counting, separation and asr," *arXiv preprint arXiv:2006.02786*, 2020.

[27] S. Sun, Y. Cheng, Z. Gan, and J. Liu, "Patient knowledge distillation for bert model compression," *arXiv preprint arXiv:1908.09355*, 2019.

[28] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu *et al.*, "Conformer: Convolution-augmented transformer for speech recognition," *arXiv preprint arXiv:2005.08100*, 2020.

[29] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, "Attention is all you need," *arXiv preprint arXiv:1706.03762*, 2017.

[30] P. Shaw, J. Uszkoreit, and A. Vaswani, "Self-attention with relative position representations," *arXiv preprint arXiv:1803.02155*, 2018.

[31] Z. Dai, Z. Yang, Y. Yang, J. Carbonell, Q. V. Le, and R. Salakhutdinov, "Transformer-xl: Attentive language models beyond a fixed-length context," *arXiv preprint arXiv:1901.02860*, 2019.

[32] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 7132–7141.

[33] S. Watanabe, T. Hori, S. Kim, J. R. Hershey, and T. Hayashi, "Hybrid ctc/attention architecture for end-to-end speech recognition," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 8, pp. 1240–1253, 2017.

[34] S. Chen, Y. Hou, Y. Cui, W. Che, T. Liu, and X. Yu, "Recall and learn: Fine-tuning deep pretrained language models with less forgetting," *arXiv preprint arXiv:2004.12651*, 2020.

[35] S. Chen, Y. Wu, Z. Chen, J. Wu, T. Yoshioka, S. Liu, J. Li, and X. Yu, "Ultra fast speech separation model with teacher student learning," *submitted for Interspeech*, 2021.

[36] T. Yoshioka, H. Erdogan, Z. Chen, and F. Alleva, "Multi-microphone neural speech separation for far-field multi-talker speech recognition," in *ICASSP*. IEEE, 2018, pp. 5739–5743.

[37] Y. Hu, Y. Liu, S. Lv, M. Xing, S. Zhang, Y. Fu, J. Wu, B. Zhang, and L. Xie, "Dccrn: Deep complex convolution recurrent network for phase-aware speech enhancement," *arXiv preprint arXiv:2008.00264*, 2020.

[38] C. Wang, Y. Wu, Y. Du, J. Li, S. Liu, L. Lu, S. Ren, G. Ye, S. Zhao, and M. Zhou, "Semantic mask for transformer based end-to-end speech recognition," *arXiv preprint arXiv:1912.03010*, 2019.