BREAKING TRADE-OFFS IN SPEECH SEPARATION WITH SPARSELY-GATED MIXTURE OF EXPERTS

Xiaofei Wang, Zhuo Chen, Yu Shi, Jian Wu, Naoyuki Kanda, Takuya Yoshioka

Microsoft, One Microsoft Way, Redmond, WA 98052, USA

ABSTRACT

Several trade-offs need to be balanced when employing monaural speech separation (SS) models in conversational automatic speech recognition (ASR) systems. A larger SS model generally achieves better output quality at an expense of higher computation, meanwhile, a better SS model for overlapping speech often produces distorted output for non-overlapping speech. This paper addresses these trade-offs with a sparsely-gated mixture-of-experts (MoE). The sparsely-gated MoE architecture allows the separation models to be enlarged without compromising the run-time efficiency, which also helps achieve a better separation-distortion trade-off. To further reduce the speech distortion without compromising the SS capability, a multi-gate MoE framework is also explored, where different gates handle non-overlapping and overlapping frames differently. ASR experiments are conducted by using a simulated dataset for measuring both the speech separation accuracy and the speech distortion. Two advanced SS models, Conformer and WavLM-based models, are used as baselines. The sparsely-gated MoE models show a superior SS capability with less speech distortion, meanwhile marginally increasing the run-time computational cost. Experimental results using real conversation recordings are also presented, showing MoE’s effectiveness in an end-to-end evaluation setting.

Index Terms— Speech separation, mixture-of-experts, automatic speech recognition, conversational transcription

1. INTRODUCTION

Speech separation (SS) is an approach to handling overlapping voices in automatic speech recognition (ASR) of conversational audio [1][2][4][5]. It provides overlap-free signals by splitting an observed audio signal into several channels, thereby allowing the ASR system to distinguish the spoken contents of different speakers even when their voices are overlapped. A great deal of effort has been made to develop advanced neural network (NN) architectures for improving the SS quality [5][6][8]. Recent studies leveraging self-attention-based models, e.g., Transformer [9] and Conformer [10], showed their superior SS capabilities. In most existing studies, the NN models were trained with a limited amount of simulated noisy-clean data pairs, preventing the models from generalizing to unseen domains [11][12]. Self-supervised learning (SSL) alleviates this problem by using a large amount of unpaired noisy data [3][14][15]. WavLM, a recently proposed SSL model, achieved the best SS accuracy by a large margin [16][17]. With these technical advancements, the SS technology has come to practical use [18].

When building SS models for production, one faces two kinds of trade-offs. The first trade-off is from the model capacity and the computational cost. While a very large SS model could be trained to achieve better output quality, simply scaling up the model capacity would result in a significant increase of inference cost [17]. The second trade-off lies between SS capability and speech distortion. More specifically, a better SS model for overlapping speech often produces worse (i.e. more distorted) output for non-overlapping speech. This trade-off becomes an issue, especially in monaural SS [10]. The speech distortions introduced by the SS processing are observed as the degradation of ASR accuracy during single-talker regions with a high signal-to-noise ratio [10][19][20][21].

This paper examines the effectiveness of sparsely-gated model architectures for dealing with these trade-offs. Specifically, to resolve the trade-off between model capacity and inference cost, we adopt Switch Transformer [22] to increase the model capacity in a computationally efficient way [23]. Switch Transformer was developed based on the Mixture-of-Experts (MoE) [24][25] framework. Each feed-forward neural network (FFN) layer of a standard Transformer or Conformer block can be replaced with multiple parallel FFN layers, or experts. In each layer, only one of the experts is activated for each input data based on an additional routing function, during both training and inference [23]. Therefore, the model capacity can be significantly enlarged with a modest extra computational cost for the routing function. We further investigate the separation-distortion trade-off with the multi-gate MoE (MMoE) architecture of [26]. With MMoE, the non-overlapping and overlapping training batches are routed to different gating networks. The effectiveness of the proposed methods is examined based on comprehensive ASR-based evaluation using two state-of-the-art base model architectures, i.e., Conformer [10] and WavLM-based dense models [17], and both simulated and real data.

2. PRIOR WORK

2.1. Sparsely-gated MoE and MMoE

A typical MoE layer consists of \( N \) expert networks \( E_i \), where \( i \in \{1, ..., N\} \), and a gating network \( G \) [24]. MoE layer output \( y \) is a weighted combination over \( m \) experts \( (m \leq N) \) as follows:

\[
y = \sum_{i \in T(G(x))} G(x), E_i(x),
\]

where \( T(\cdot) \) is a function to return indices of top-\( m \) experts given \( G(x) \in \mathbb{R}^N \). \( G \) calculates Softmax over the product of input \( x \) and a trainable router matrix such that \( G(x) \) represents the gating probability for the \( i \)-th expert \( E_i \). Fedus et al. [22] proposed Switch Transformer to further reduce MoE’s computational cost by using a switching approach, which routes \( x \) to only one expert. This is done by sparsifying \( G \)’s output with a one-hot activation, where the indexes of the experts with lower Softmax values are masked. The switching MoE mechanism can be used in Transformer and Conformer (or dense models) by replacing each FFN layer of these mod-
els by an MoE layer consisting of \( N \) expert FFN layers of the equivalent size. This significantly increases the model capacity with a very marginal run-time cost increase needed for the gating function computation.

To train the MoE models, an auxiliary loss function is often applied to balanced data loads across the experts. This is needed to prevent the input data from always being routed to a limited number of specific experts and thus the other experts from underutilized. The auxiliary MoE loss, \( L_{\text{MoE}} \), is given by \cite{22}:

\[
L_{\text{MoE}} = \alpha N \sum_{i=1}^{N} f_i P_i,
\]

where \( f_i \) is the fraction of the training dispatched to \( E_i, P \), the fraction of the router probability allocated for the \( i \)-th expert, and \( \alpha \) is the auxiliary loss weight.

MMoE is an extension of MoE, which was originally designed for multi-task learning \cite{26}. It uses multiple gating networks optimized for individual tasks while sharing the experts for all the tasks. However, in this work, we only have a single SS objective, the usage of MMoE will be introduced in Section 3.4.

2.2. Monaural speech separation (SS)

Speech signals of natural human conversations occasionally contain overlapped voices from different speakers, which negatively impacts the ASR accuracy for each person. SS uses NNs to recover individual speaker signals even under noisy and reverberant conditions. In this work, we focus on monaural SS models with a frequency domain input. Specifically, the SS model takes in the short-time Fourier transform (STFT) magnitude features, \( Y \in \mathbb{R}^{T \times F} \), derived from the observed signal and outputs time-frequency (T-F) masks \( M_i \in \mathbb{R}^{T \times F} \) for each speaker \( i \) as follows.

\[
M_i = \text{SS}(Y; \theta), i = 1, \ldots, S,
\]

where \( \theta \) represents the parameter set of the SS model. The \( i \)-th speech signal is estimated by computing the product of \( M_i \) and \( Y \) and applying inverse STFT. The SS model is optimized by an utterance-level permutation invariant training (uDPI) loss as follows.

\[
L_{\text{UPI}} = \min \sum_{(i,j) \in P} \| \text{Mel}(M_i \odot Y) - \text{Mel}(X_j) \|_2^2, \tag{4}
\]

where \( P \) is the set of all possible permutations of the \( S \) speech sources, \( \odot \) denotes element-wise multiplication for each T-F bin, \( |X_j| \) is the magnitude of the \( j \)-th reference speech, and \( \text{Mel}() \) is the mel-filterbank transform function.

Among the successful approaches, we choose the Conformer structure \cite{5} (Fig. 1(a)) consisting of a stack of Conformer blocks as our backbone SS model estimation model. The number of maximum simultaneously active speakers, \( S \), was set to 2 in our experiments, which covers most meeting transcription scenarios.

The SS model can be applied to continuous speech separation (CSS) to deal with a long-form conversational audio signal with streaming processing \cite{4,5,10}. CSS is realized by using a sliding window (e.g., 2.4-second block with 0.8-second hop), where each windowed signal is processed with the SS model locally and the output signals are stitched across the windows \cite{10}.

2.3. WavLM-based SS

In a spirit of building on top of the state of the art, we also examine the effect of the sparsely-gated MoE approach on a WavLM-based SS model, which we briefly review below. SSL techniques, including WavLM, allow a large amount of unpaired noisy data to be leveraged for model training and has shown SS performance improvement \cite{14,15,27}.

In this paper, we follow the strategy which we have identified in our concurrent work \cite{17} for leveraging WavLM in SS. For each input sample, layer-wise speech representations are extracted from a pretrained WavLM model. Then, their weighted average is calculated with layer-wise learnable weights and concatenated with the STFT feature. The new T-F mask estimation is reformulated as

\[
M_i = \text{SS}(% (\text{Concat}(\text{WavLM}_m(Y, Y); \theta), i = 1, \ldots, S. \tag{5}
\]

where \( \text{Concat}(\cdot) \) denotes the weighted average of embeddings using the first \( n \) layers of the WavLM model, and Concat(\cdot) feature concatenates \cite{15,16,27}. Note that only the first \( n \) layers are used because these layers were found to have a large impact in SS \cite{15} while this also helps moderate the computational cost increase resulting from the WavLM model computation.

3. METHOD

In this section, we introduce how MoE is applied to SS model to increase the model capacity without significantly increasing the inference cost. We further discuss reducing the speech processing distortion—another important trade-off in SS—by leveraging the MMoE architecture.

3.1. Conformer-based SS with MoE

Fig. 1 shows the overview of our Conformer-based SS model with MoE or MMoE. We use a variant of Conformer block as shown in Fig. 1(b). Our Conformer block consists of a multi-head self-attention module with relative position encoding, a convolution module, and a dense FFN module, where each module has a residual connection and layer normalization. As with \cite{5}, no dropout layers are included within the Conformer block.

For introducing MoE into the Conformer-based SS, the dense FFN module in a Conformer block is replaced by an MoE module as depicted in Fig. 1(c). Each expert in the MoE module consists of a linear layer, a ReLU activation, a dropout layer, and another linear layer. The shapes of the linear layers remain the same as those of the dense FFN module such that the computational cost related to the FFN layers becomes identical to the model with MoE and that without MoE. We apply the MoE module for every other Conformer block starting from the bottom block and always use the top-1 routing gate for expert selection, in both training and inference.
The loss function $L_{\text{SS-MoE}}$ for the SS with MoE is based on $L_{\text{MoE}}$ (Eq. (2)) and $L_{\text{uPIT}}$ (Eq. (4)) as,

$$L_{\text{SS-MoE}} = L_{\text{MoE}} + L_{\text{uPIT}}.$$  

(6)

3.2. Conformer-based SS with MMoE

Prior studies [19][21] showed that a better trade-off between separation quality and speech distortion could be achieved by introducing a learnable switching module to determine whether SS is applied or not depending on the input signal. These studies motivate us to examine the application of different gating functions for overlapping and non-overlapping speech. More specifically, we hypothesize that we can promote some experts to focus on the processing of non-overlapping speech with less speech distortion if we introduce a gating function specialized for non-overlapping speech.

To this end, we apply MMoE with two gating functions for SS model (Fig. 1(c)). During the training, either of two gating functions is used depending on whether the training audio batch consists of speech overlaps or not. For example, depicted by Fig. 1(c), when the non-overlapping batch presents while iterating the data loader for training, gate A is active and gate B is disabled; when the overlapping batch presents, it is the other way around. Note that we grouped the overlapping or non-overlapping audio samples into different training batches in advance. We apply voice activity detection for reference signals to determine the existence of speech overlaps. In inference, the gating function used for non-overlapping speech in training is always activated with the other deactivated, which was found to be effective in our experiment. As same with the Conformer-based SS with MoE, we apply the MMoE layer for every other Conformer block starting from the bottom block.

4. EXPERIMENTAL RESULTS

We experimentally evaluated the effectiveness of the proposed MoE-based SS model architecture in terms of ASR accuracy, with and without a pre-trained WavLM-based SSL model. Both utterance-based SS and CSS were considered by using an artificially mixed evaluation dataset and real evaluation corpus, respectively. The former enabled directly assessing the benefits of the sparsely-gated architectures under different SS settings while the latter shed light on the usefulness in an end-to-end setting. This section describes the experimental setups and evaluation results with our analyses.

4.1. Experimental setups

4.1.1. Evaluation

Our artificially mixed evaluation dataset consisted of single-talker utterances (non-overlapping data) and their artificial mixtures (overlapping data). These utterances were sampled from the single-talker regions of our in-house meeting recordings [17]. The original utterances were grouped into close-talking and far-field speech data sets by the data collecting settings. The total amounts of the data were approximately 100k and 300k words, respectively. Accordingly, we ended up with the following four subsets for the ASR experiment.

- **Close Clean**: non-overlapping utterances recorded with close-talk microphone
- **Far Clean**: non-overlapping utterances recorded with far-field microphone
- **Close Mix**: overlapping utterances simulated by Close Clean with a 40% overlap ratio
- **Far Mix**: overlapping utterances simulated by Far Clean with a 40% overlap ratio

Close Mix and Far Mix were generated by adding two random utterances with random delays, which were used to measure the separation accuracy. They had the same numbers of words as Close Clean and Far Clean, respectively. These non-overlapping data sets were used to measure the effects of the processing artifacts caused by the SS models. ASR was performed for each separate signal with a hybrid model [18]. The WERs were calculated with the best permutation between the hypotheses obtained from the separated signals and the reference transcriptions.

For the CSS real evaluation, we used the ICSI [28] and AMI [29] single-distant-microphone (SDM) evaluation sets with utterance-group segmentations [30]. The speaker-agnostic WER was used to measure both the separation accuracy and the speech distortions. In addition, we measured the real-time factors (RTF) using an Intel Xeon W-2133 3.60GHz Processor with the single-thread setup. The RTF was measured by running each model on a 2.4-second sample 100 times and taking the average of the wall-clock times.

4.1.2. Configurations of SS Models and MoE

To examine the generalizability of the proposed approach, we employed two Conformer-based SS model configurations [5]. The first configuration used a 257-dimension STFT (denoted by $|\text{STFT}|$) for each frame as input. This model had 18 Conformer encoder layers, each consisting of an 8-head and 512-dimension self-attention layer, a 1024-dimension FFN layer, and a convolution layer with kernel size 33 and channel number 512. The numbers of the parameters were 59M.

The second configuration utilized WavLM-based embeddings as additional features. To keep the total computational cost within a reasonable range, the SS model of the second type was smaller (26M parameters), containing 16 Conformer layers, each with 4 attention heads and 256 dimensions. The other model parameters were the same as those of the first model, except that the additional feature dimension from WavLM was 1,024, resulting in an 1,281-dimension input feature for this SS model. The WavLM model consisted of 24 layers with 317M of parameters, in which the bottom $n = 8$ layers consisting of 103M parameters were used for feature extraction, denoted by WavLMs. For more details, see section 2.3 and [17].

In case of MoE-based model, experts were equally distributed to GPUs. Capacity factor [22] and expert dropout rate were set to 1.5 and 0.1, respectively. The switching jitters, a variable controlling the multiplicative fraction to the gate input, and $\alpha$ were set to 0.01 [31].

4.1.3. Training

Training schemes: The two SS model configurations described above used different training procedures as follows. The STFT-based models were trained from scratch in a single stage by using 10k hours of simulated data. On the other hand, the WavLM-based models were trained in two or three stages, following the best recipe of [17]. Specifically, in the first stage, the model was initialized by leveraging a smaller 7k-hour simulated data set, followed by training using the 10k-hour data in the second and third stages. In each stage, the model parameters were initialized by the last check-point of the previous training stage. The WavLM model parameters were frozen in the first two stages while they were fine-tuned in the third stage.

Data: Both the 10k and 7k hours of data were artificially generated based on the recipe of [8]. Each training dataset consisted of a balanced combination of single-speaker utterances and two-speaker mixtures, with four kinds of overlap patterns [5]. The difference between two datasets lies in the clean sources used for the data simulation. The 7k training dataset used Voxelab [32] while the 10k dataset used various sources including WSJ, LibriSpeech and our internal datasets. The image method [33] was used to generate the room impulse response. The background noise samples were taken from the DNS challenge [34] and MUSAN noise datasets [35]. We used 4 second of short segmented audio for the model training.

Configuration details: All models were trained with eight NVIDIA A100 GPUs and an AdamW optimizer using a weight decay of
Table 1. WERs (%) of dense and MoE-based SS models for utterance-based evaluation.

| Features | Model | Number of Experts | RTF | WER (%) |
|----------|-------|-------------------|-----|---------|
|          |       |                   |     | Close Mix | Clean | Far Mix | Clean |
| No Pro. |       | -                 | -   | 91.9      | 12.8  | 86.8    | 13.5  |
| [STFT]  | Dense | (59M)             | 0.130 | 23.5    | 20.9  | 23.0    | 15.0  |
|         | MoE   | 4 (67M)           | 0.132 | 21.9    | 17.3  | 21.9    | 14.9  |
|         | MoE   | 8 (125M)          | 0.136 | 21.4    | 17.7  | 22.7    | 15.0  |
|         | MoE   | 16 (201M)         | 0.141 | 21.5    | 17.6  | 22.1    | 15.1  |
| [STFT] + WavLMs | Dense | (26M)           | 0.310 | 21.7    | 20.9  | 21.6    | 14.9  |
|         | MoE   | 4 (43M)           | 0.313 | 18.1    | 17.5  | 20.5    | 15.1  |
|         | MoE   | 8 (60M)           | 0.316 | 17.9    | 17.0  | 20.2    | 15.2  |
|         | MoE   | 16 (77M)          | 0.321 | 20.3    | 17.3  | 22.6    | 15.1  |

† We only show the number of parameters for the SS part. The models were trained based on the first two stages in the three-stage training.

4.2. Results of utterance-based evaluation

4.2.1. Effects of MoE

Table 1 lists the WER results of the utterance-based evaluation experiments. The first row, labeled as No Pro., show the WERs of the unprocessed mixtures (Close Mix and Far Mix) and single-talker signals (Close Clean and Far Clean). They reveal the detrimental effects of speech overlap and the ASR performance for the distortion-free signals, respectively. The dense model using the [STFT] feature yielded significant WER improvements over No Pro. for the mixture signals. However, this was obtained at the expense of increased WERs for the non-overlapping signals, where the WER of Close Clean set was severely degraded from 12.8% to 20.9%. A similar issue is observed for the WavLM-based model ([STFT] + WavLMs), while it consistently outperformed the STFT-based model.

The upper half of Table 1 compares the dense SS model with 59M parameters with MoE-based models using different numbers of experts for the STFT-based features. It can be clearly seen that all the MoE-based models outperformed the dense model, showing the performance benefit of sparsely gated architectures. It should be noted that the computational costs of these models were almost the same, with the relative RTF difference being in the range of 1.5%-8.5%. The performance improvement saturated with four experts for the [STFT]-based SS model. It should also be noted that the MoE models significantly reduced the WERs in the Close Mix and Close Clean sets, implying less speech distortion under the high SNR conditions. A similar trend can be observed in the WavLM-based experiments shown by the lower half of Table 1. The 8-expert MoE setup was found to be best for the WavLM-based SS model using [STFT] + WavLMs. These results suggest that the MoE-based framework is effective in enhancing the model capacity without increasing the runtime cost and that it provided a better compromise regarding the separation-distortion trade-off.

4.2.2. Effects of MMoE

Table 2 shows the comparison results of the MMoE-based models with the MoE counterparts. Based on the WER results in Table 1, we used 4 and 8 experts for the STFT and WavLM-based models, respectively. For both STFT and WavLM-based models, using MMoE with oracle gates considerably improved the WERs for the clean conditions while retaining the overall performance for the mix condition, suggesting that they produced less distorted output signals. This supports our hypothesis that the multi-gate approach further promotes some experts being optimized for dealing with non-overlapping speech to suppress the speech distortion. This hypothesis was further validated by another experiment where we always used the non-overlapping gate (see the rows with “No” in the “Oracle Gate” column). This degraded the WERs for the mix conditions. Interestingly, the level of the degradation was limited especially with the STFT input probably because our model interleaved the dense and MMoE Conformer blocks with the former providing the separation capability even for the non-oracle condition. With WavLM-model input, the WER gap between the oracle and non-oracle gate settings was larger while the MMoE-based model still outperformed the dense model, especially for the clean conditions. This is beneficial for scenarios with a relatively low overlapping ratio, such as one-on-one conversations. Further, benefiting from MMoE to the fullest extent would require a low-cost overlap-detection module, which will be explored in the future.

4.3. CSS evaluation and computational cost analysis

Table 3 shows the WERs of selected models in the CSS-based evaluation. We chose the 8-expert MoE-based model in the WavLM setting based on the far-field results of Table 2. The WER gains were consistently observed for both ICSI and AMI in this end-to-end evaluation setting. The RTF difference between the two models was less than 0.02% despite the significant increase in the parameter number, suggesting the MoE’s capability to mitigate the model-size-performance trade-off of the conventional dense models.

5. CONCLUSIONS

In this work, we present the investigations of applying the sparsely-gated architectures including MoE and MMoE for the SS task. Extensive evaluations on both the dense and MoE-supported models had been conducted by using utterance-based and CSS datasets. ASR results and RTF analysis showed that the integration of the sparsely gated MoE architecture can efficiently increase the capacity of the model which outperformed the dense ones in terms of both separation accuracy and speech distortion, meanwhile retaining a low computational cost.
6. REFERENCES

[1] I. Himawan, I. McCowan, and M. Lincoln, “Microphone array beamforming approach to blind speech separation,” in Machine Learning for Multimodal Interaction. Springer, 2007, pp. 295–305.

[2] J. McDonough, K. Kumatani, T. Gehrig et al., “To separate speech,” in Machine Learning for Multimodal Interaction. Berlin, Heidelberg: Springer, 2007, pp. 283–294.

[3] T. Yoshioka, H. Erdogan et al., “Recognizing overlapped speech in meetings: A multichannel separation approach using neural networks,” Proc. Interspeech, pp. 3038–3042, 2018.

[4] D. Raj, P. Denisov, Z. Chen et al., “Integration of speech separation, diarization, and recognition for multi-speaker meetings: System description, comparison, and analysis,” in Proc. SLT, 2021, pp. 897–904.

[5] J. Wu, Z. Chen, S. Chen et al., “Investigation of practical aspects of single channel speech separation for asr,” in Proc. Interspeech, 2021, pp. 3066–3070.

[6] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, “Deep clustering: Discriminative embeddings for segmentation and separation,” in Proc. ICASSP, 2016, pp. 31–35.

[7] D. Yu, M. Kolbæk et al., “Permutation invariant training of deep models for speaker-independent multi-talker speech separation,” in Proc. ICASSP, 2017, pp. 241–245.

[8] Y. Luo, Z. Chen, and T. Yoshioka, “Dual-path rn: efficient long sequence modeling for time-domain single-channel speech separation,” in Proc. ICASSP, 2020, pp. 46–50.

[9] C. Subakan, M. Ravaneli, S. Cornell, M. Bronzi, and J. Zhong, “Attention is all you need in speech separation,” in Proc. ICASSP, 2021, pp. 21–25.

[10] S. Chen, Y. Wu, Z. Chen, J. Wu, J. Li, T. Yoshioka, C. Wang, S. Liu, and M. Zhou, “Continuous speech separation with conformer,” in Proc. ICASSP, 2021, pp. 5749–5753.

[11] T. Yoshioka, X. Wang, D. Wang et al., “VarArray: Array-geometry-agnostic continuous speech separation,” in Proc. ICASSP, 2022, pp. 6027–6031.

[12] X. Wang, D. Wang, N. Kanda et al., “Leveraging real conversational data for multi-channel continuous speech separation,” in Proc. Interspeech, 2022, pp. 3814–3818.

[13] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, “wav2vec 2.0: A framework for self-supervised learning of speech representations,” Advances in Neural Information Processing Systems, vol. 33, pp. 12 449–12 460, 2020.

[14] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai et al., “Hubert: Self-supervised speech representation learning by masked prediction of hidden units,” IEEE/ACM Trans. on Audio, Speech, and Language Processing, vol. 29, pp. 3451–3460, 2021.

[15] S. Chen, C. Wang, Z. Chen et al., “WavLM: Large-scale self-supervised pre-training for full stack speech processing,” IEEE Journal of Selected Topics in Signal Processing, 2022.

[16] X. Chang, T. Maekaku, Y. Fujita, and S. Watanabe, “End-to-end integration of speech recognition, speech enhancement, and self-supervised learning representation,” in Proc. Interspeech, 2022, pp. 3819–3823.

[17] Z. Chen and et al., “Speech separation with large-scale self-supervised learning,” in Submitted to ICASSP 2023, 2022.

[18] T. Yoshioka, I. Abramovski, C. Aksoyler et al., “Advances in online audio-visual meeting transcription,” in Proc. ASRU, 2019, pp. 276–283.

[19] H. Sato, T. Ochiai, M. Delcroix, K. Kinoshita, T. Moriya, and N. Kamo, “Should we always separate?: Switching between enhanced and observed signals for overlapping speech recognition,” in Proc. Interspeech, 2021, pp. 1149–1153.

[20] K. Iwamoto, T. Ochiai, M. Delcroix, R. Ikehata, H. Sato, S. Araki, and S. Katagiri, “How bad are artifacts?: Analyzing the impact of speech enhancement errors on asr,” in Proc. Interspeech, 2022, pp. 5418–5422.

[21] H. Sato, T. Ochiai, M. Delcroix et al., “Learning to enhance or not: Neural network-based switching of enhanced and observed signals for overlapping speech recognition,” in Proc. ICASSP, 2022, pp. 6287–6291.

[22] W. Fedus, B. Zoph, and N. Shazeer, “Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity,” Journal of Machine Learning Research, vol. 23, no. 120, pp. 1–39, 2022.

[23] N. Shazeer, A. Mirhoseini, K. Maziarz et al., “Outrageously large neural networks: The sparsely-gated mixture-of-experts layer,” arXiv preprint arXiv:1701.06538, 2017.

[24] R. A. Jacobs, M. I. Jordan, S. J. Nowlan, and G. E. Hinton, “Adaptive mixtures of local experts,” Neural computation, vol. 3, no. 1, pp. 79–87, 1991.

[25] M. I. Jordan and R. A. Jacobs, “Hierarchical mixtures of experts and the EM algorithm,” Neural computation, vol. 6, no. 2, pp. 181–214, 1994.

[26] J. Ma, Z. Zhao, X. Yi, J. Chen, L. Hong, and E. H. Chi, “Modeling task relationships in multi-task learning with multi-gate mixture-of-experts,” in Proc. KDD, 2018, pp. 1930–1939.

[27] Z. Huang, S. Watanabe, S. Yang et al., “Investigating self-supervised learning for speech enhancement and separation,” in Proc. ICASSP, 2022, pp. 6837–6841.

[28] A. Janin, D. Baron, J. Edwards et al., “The icsi meeting corpus,” in Proc. ICASSP, vol. 1, 2003, pp. I–I.

[29] J. Carletta, S. Ashby, S. Bourban et al., “The ami meeting corpus: A pre-announcement,” in International workshop on machine learning for multimodal interaction. Springer, 2005, pp. 28–39.

[30] N. Kanda, G. Ye et al., “Large-scale pre-training of end-to-end multi-talker asr for meeting transcription with single distant microphone,” in Proc. Interspeech, 2021, pp. 3430–3434.

[31] [Online]. Available: https://github.com/pytorch/ort/tree/main/ori_moc

[32] A. Nagrani, J. S. Chung, W. Xie, and A. Zisserman, “Voxceleb: Large-scale speaker verification in the wild,” Computer Speech & Language, vol. 60, p. 101027, 2020.

[33] J. B. Allen and D. A. Berkley, “Image method for efficiently simulating small-room acoustics,” The Journal of the Acoustical Society of America, vol. 65, no. 4, pp. 943–950, 1979.

[34] C. K. Reddy, H. Dubey, V. Gopal et al., “ICASSP 2021 deep noise suppression challenge,” in Proc. ICASSP, 2021, pp. 6623–6627.

[35] D. Snyder, G. Chen, and D. Povey, “MUSAN: A music, speech, and noise corpus,” arXiv preprint arXiv:1510.08484, 2015.