INTERSPEECH 2020
October 25–29, 2020, Shanghai, China

Learning Better Speech Representations by Worsening Interference

Jun Wang

Tencent AI Lab, China
joinerwang@tencent.com

Abstract

Can better representations be learnt from worse interfering scenarios? To verify this seeming paradox, we propose a novel framework that performed compositional learning in traditionally independent tasks of speech separation and speaker identification. In this framework, generic pre-training and compositional fine-tuning are proposed to mimic the bottom-up and top-down processes of a human's cocktail party effect. Moreover, we investigate schemes to prohibit the model from ending up learning an easier identity-prediction task. Substantially discriminative and generalizable representations can be learnt in severely interfering conditions. Experiment results on downstream tasks show that our learnt representations have superior discriminative power than a standard speaker verification method. Meanwhile, RISE achieves higher SI-SNRi consistently in different inference modes over DPRNN, a state-of-the-art speech separation system.

Index Terms: cocktail party problem, speech separation, speaker identification

1. Introduction

Deep speaker embedding has successful applications in modern systems [1, 2, 3] for speaker identification (SI), speaker verification (SV), and speaker diarization (SD) tasks. These systems generally require complicated pipelines. Before learning the speaker embedding, a speech activity detection (SAD) module and a segmentation module are needed to generate short speech segments with no interference or overlapping; after extracting the speaker embedding, a clustering module is needed to group the short segments to corresponds to one speaker identity. If the system is required to deal with overlapped speech or interference, a detector and classifier module is also needed in the pipeline to remove the overlapped segments. However, still, performances of standard systems are remarkably hurt in highly overlapped scenarios [4]. To simplify these complicated procedures, [5] recently proposed a new approach to combine the SAD, segmentation and embedding extraction into one stage.

"Cocktail party problem"[6] is another important research topic developed individually and separately from the above. Recent advances of deep-learning speech separation models have drastically advanced the state-of-the-art performances on several benchmark datasets. Currently, best-performing solutions are mainly based on a time-domain audio separation network (TasNet) [7], which has achieved ground-breaking speech separation performance by directly applying PIT [8, 9] on waveforms. Variants of TasNet models are subsequently proposed: First, TasNet emerged with Bi-LSTM based separation network [7] is proposed, and surpassed by TCNs [10, 11] with substantial performance gain. Until very recently, the record of TCNs has been further advanced by dual-path RNNs (DPRNNs) [12]. DPRNNs' record was recently surpassed by [13] by adding speaker stacks jointly trained with the separation stack. However, it is computationally much more expensive than DPRNNs due to the K-means clustering to infer speaker embeddings [14].

Neurobiology study [15] describes the cocktail party effect that listeners may immediately detect words from (either attended or not) acquaintance in highly interfering conditions, for instance hearing a friend calling one's name among a wide range of auditory input. In this process, the human auditory system does not perform speaker identification and speech separation or extraction task separately but instead follows a bottom-up and top-down processes [16]. Recent studies [17, 18, 19, 20] strive to mitigate the limit of supervised models via unsupervised and self-supervised learning. Inspired by the above, we propose a novel framework with the following contribution.

On the one hand, we model the bottom-up process by first pre-training the model for a generic intrinsic task. The purpose is to extract generic representations with separability from wave bits of masked speech with interference. Then part of the model is fine-tuned towards down-stream tasks of identification and separation. On the other hand, we try to model the top-down process during fine-tuning, that the extracted high-level abstract representations (speaker vectors here) are fed back to modulate the model's behavior on the low-level bit-wise separation task. Our framework is free from K-means clustering, SAD, segmentation, overlap detector nor necessarily offline enrollment, and allows for more efficient training than [13]. Meanwhile, we also indicate an important factor that prohibits the model from ending up learning an easier identity-prediction task during training. Consequently, our model is enforced to learn substantially more discriminative and generalizable representations.

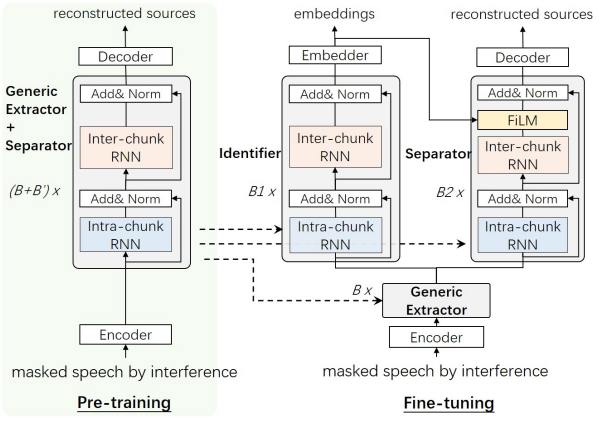

Figure 1: *Procedures in RISE: (left) Pre-training phase for the intrinsic task of learning feature with discriminative power between speech and interference, and (right) fine-tuning phase for downstream tasks.*

2. Model Architecture

We introduce a framework for learning Representations via Interference Separation and Extraction (RISE). It has a unified ar-

http://dx.doi.org/10.21437/Interspeech.2020-1545

chitecture across different tasks, as shown in Fig. 1, where the overall procedures include pre-training and fine-tuning steps. The purpose of the intrinsic task in pre-training is to learn a generic feature space with discriminative power between speech and interference. In fine-tuning, the generic modules can serve as feature extraction and freeze from fine-tuning. Each down-stream task can share these generic modules and only have the task-specific modules fine-tuned. This structure design enables efficient training. For comparison purposes, RISE was chosen to ensemble the same basic block as those of DPRNN.

2.1. Pre-training RISE

2.1.1. Intrinsic task: masked speech model

Similar to the concept of "masked language model" (MLM) [21, 22], RISE uses a "masked speech model" (MSM) pre-training objective. The MSM masks the speech by mixing it with random interference. Different from the artificial MLM in [22], MSM simulates the "Cocktail party" problem in reality. To simulate the variety in realistic scenarios, we mask the speech with different random interference speakers at random blurt-out positions for different training epochs, and the signal-to-interference ratio (SIR) is randomly sampled from 0 to 5dB.

2.1.2. Modules and notations

For comparison purposes, the Encoder and the Decoder have the same structure and size as in [11]; the Generic Extractor and the Separator ensemble the same basic block architectures of DPRNN [1] block in [12].

The **Encoder** transforms a waveform mixture of C sources, $\sum_{c=1}^{C}\mathbf{s}_c$ being masked speech with interference, into a sequential input $\mathbf{W}\in\mathbb{R}^{N\times L}$, where N is the feature dimension, and L is the time step number. Then $\mathbf{W}$ is segmented to form a 3-D tensor $\mathbf{T}\in\mathbb{R}^{N\times K\times S}$, where K is the chunk size, and S is the number of chunks.

Then in the **Generic Extractor** and **Separator**, the 3-D tensor is passed to the stacks of $B+B'$ DPRNN blocks. Each block contains an intra-chunk and inter-chunk RNN processing, each of which transforms the input 3-D tensor $\mathbf{T}_b$ into another tensor with the same shape $\bar{\mathbf{T}}_b\in\mathbb{R}^{N\times K\times S}$, and the input and output tensors are then passed to an Add& Norm residual connection. The output of each block is served as the input $\mathbf{T}_{b+1}$ to the next block.

In the projector, the output of the last DPRNN block $\mathbf{T}_{B+B'+1}\in\mathbb{R}^{N\times K\times S}$ is passed to a non-linear mapping (we use a parametric rectified linear unit PReLU) and a 2-D convolutional layer to project to a higher-dimension feature space of $(N\times C)$, corresponding to C sources. Now we have $\mathbf{T}_o=\text{Conv2D}(\text{PReLU}(\mathbf{T}_{B+B'+1}))\in\mathbb{R}^{(N\times C)\times K\times S}$.

Then, the 3-D tensor is merged back (as opposed to the above chunking process) to C sequences: $\bar{\mathbf{Q}}_c\in\mathbb{R}^{N\times L}, c=1,..C$. These sequences are then passed through a gated output layer, which consists of a 1-D convolutional and a non-linear layer, to compute C estimated masks: $\mathbf{Q}_c=\text{Tanh}(\text{Conv1D}(\bar{\mathbf{Q}}_c))\odot\text{Sigmoid}(\text{Conv1D}(\bar{\mathbf{Q}}_c))$.

Finally, in the **Decoder**, C waveform outputs are reconstructed by applying the estimated masks to the mixture input and then the overlap-add operation: $\hat{\mathbf{s}}_c=\text{OverlapAdd}(\text{Linear}(\mathbf{W}\odot\text{ReLU}(\text{Conv1D}(\mathbf{Q}_c))))$.

[1] https://github.com/ShiZiqiang/dual-path-RNNs-DPRNNs-based-speech-separation

2.1.3. Pre-training objective

The task in pre-training is to learn a generic feature space with separability power among different sources in the mixture. The representations are actually learnt only from the sources themselves by masking with each other, but not from any label; thus this is essentially a self-supervised process. We use utterance-level permutation invariant training (u-PIT) [8] to calculate the scale-invariant signal-to-noise ratio (SI-SNR) [23] as the pre-training objective:

$$\mathcal{L}=\min_{\{j\}_{1,..,A_C^C}}\sum_{i=1}^{C}\mathcal{L}_{\text{SI-SNR}}(\mathbf{s}_i,\hat{\mathbf{s}}_j),\tag{1}$$

where $\{j\}_{1,..,A_C^C}$ is full permutation of the estimated sources.

2.2. Fine-tuning RISE

After pre-training, the **Generic Extractor** can serve as a feature extraction module and can spare from fine-tuning. The remaining modules start from pre-trained parameters and are fine-tuned towards down-stream tasks.

2.2.1. Fine-tuning objectives for down-stream tasks

Identifier: In this fine-tuning, we assume the speaker identity labels, composing a dictionary of G tokens, are accessible for the training data. The deep feature $\mathbf{T}_{B+1}$ extracted via the Generic Extractor is passed to the Identifier and then the projector. Afterward, the output $\mathbf{\Gamma}_o$ is pooled in the Embedder to generate a coarser-scale embedding tensor $\bar{\mathbf{\Gamma}}\in\mathbb{R}^{N\times C\times S}$ by averaging over each chunk, as well as an utterance-level embedding tensor $\widehat{\mathbf{\Gamma}}\in\mathbb{R}^{N\times C}$ by averaging over all chunks.

We design the training objective to encourage learning separable and discriminative speaker embeddings:

$$\begin{aligned}\mathcal{L}_{\text{ID}}=\min_{\{j\}_{1,..A_C^C}}&(\sum_{k=1}^{C}\mathcal{L}_{\cos}(\mathbf{E}_{i_k},\bar{\mathbf{\Gamma}}_j)+\log\sum_{g=1}^{G}e^{-\mathcal{L}_{\cos}(\mathbf{E}_g,\bar{\mathbf{\Gamma}}_j)})\\&+\frac{1}{\gamma C}\sum_{k=1}^{C}\min_{g=1,..,G;i\neq g}\log\|\mathbf{E}_{i_k}-\mathbf{E}_g\|_1,\end{aligned}\tag{2}$$

where $\mathbf{E}\in\mathbb{R}^{N\times G}$ are the target embeddings of all training speakers, and $\mathcal{L}_{\cos}$ is cosine similarity loss with a learnable scale α and bias β. The first term[2] in $\mathcal{L}_{\text{ID}}$ encourages the output embeddings $\bar{\mathbf{\Gamma}}_j$ to be close to the corresponding target embeddings $\mathbf{E}_{i_k}$; the second serves as a normalization term, which computes the overall cosine similarity between each output embedding and all the training target embeddings; the third is a regularization term to avoid collapsing to a trial solution of all zeros. γ is a weighting factor of the regularization term.

Separator: The separator receives from the above Identifier task the conditioning input: $\{\widehat{\mathbf{\Gamma}}_j\}$, where $\{j\}=\text{argmin}\mathcal{L}_{\text{ID}}$. In this compositional task, advanced methods such as [24] can be used in a framework based on self-attention blocks []. Since this framework adopts DPRNN blocks, we apply a relatively straightforward feature-wise linear modulation (FiLM) method [25], and modulate the intermediate tensors $\bar{\mathbf{T}}_b$ from each Inter-chunk RNN layer with $\{\widehat{\mathbf{\Gamma}}_j\}$:

$$\bar{\mathbf{T}}'_b=f(\widehat{\mathbf{\Gamma}}_j)\bar{\mathbf{T}}_b+h(\widehat{\mathbf{\Gamma}}_j), j\in\{j\}\tag{3}$$

[2] We omit some dimension indexes of $\bar{\mathbf{\Gamma}}$ and $\mathbf{E}$ for simpler notation.

where $f(\cdot)$ and $h(\cdot)$ are learnable functions such as neural networks (e.g., simple full connections in this work).

Till now, we have got C modulated intermediate tensors $\bar{\mathbf{T}}'_{b,i}, i = 1, ..., C$ in each DPRNN block, and thus can compute corresponding estimated sources $\hat{\mathbf{s}}_i, i = 1, ..., C$. Consequently, the calculation of the fine-tuning objective becomes PIT-free: $\mathcal{L}_{\text{sep}} = \sum_{i=1}^{C} \mathcal{L}_{\text{SI-SNR}}(\mathbf{s}_i, \hat{\mathbf{s}}_i)$.

2.2.2. Learning generalizable representations

Despite the popularity of learning token/ID embeddings in both NLP[22] and speech domain [13], we investigate three schemes to bring up a crucial argument that affects if the model succeeds (or fails) on generalizability.

Algorithm 1 Representation learning in compositional tasks of identification and separation

Input: The deep features $\mathbf{T}_{B+1}$ extracted from the Generic Extractor. Initialized Identifier and Separator model parameters $\Theta_{\mathbf{ID}}$ and $\Theta_{\mathbf{sep}}$ from pre-training. Hyperparameter λ, ε, and learning rate μ.
Initialized target embedding parameters $\mathbf{E}$ following one of the below schemes
#1: a sparse embedding lookup table that stores the target embeddings of the fixed dictionary of speaker ids.
#2: a parameter tensor, registered as learnable in the Embedder graph.
#3: a parameter tensor, detached from the Embedder graph.
Output: $\mathbf{E}$, $\Theta_{\mathbf{ID}}$, and $\Theta_{\mathbf{sep}}$.

1: **while** not converge **do**
2: Compute the joint loss by $\mathcal{L}_{\text{ID\_sep}} = \mathcal{L}_{\text{sep}} + \lambda \mathcal{L}_{\text{ID}}$
3: Update the parameters with corresponding backpropagation error: $\Theta_{\text{sep}} \leftarrow \Theta_{\text{sep}} - \mu \frac{\partial \mathcal{L}_{\text{sep}}}{\partial \Theta_{\text{sep}}}$, $\Theta_{\text{ID}} \leftarrow \Theta_{\text{ID}} - \mu \frac{\partial \mathcal{L}_{\text{ID\_sep}}}{\partial \Theta_{\text{ID}}}$. Update the embedder parameters,
IF scheme #1 or #2: $\mathbf{E} \leftarrow \mathbf{E} - \mu \frac{\partial \mathcal{L}_{\text{ID\_sep}}}{\partial \mathbf{E}}$,
IF scheme #3: $\mathbf{E}_{i_k} \leftarrow (1-\varepsilon)\mathbf{E}_{i_k} + \varepsilon \widehat{\mathbf{\Gamma}}_j$, where $\{j\} = \text{argmin}\mathcal{L}_{\text{ID}}$
4: Schedule the learning rate μ.
5: **end while**

2.2.3. Inference

During inference for the speech separation or extraction task, our RISE framework has the flexibility to apply different modes: an "online" mode and a "guided" mode. In the "online" mode, the speaker embeddings $\widehat{\mathbf{\Gamma}}_j$ are estimated and be applied online following Eq. 3; in the guided mode, we assume $\widehat{\mathbf{\Gamma}}_j$ in Eq. 3 has been pre-computed using some enrollments from the speaker and does not need to be estimated online.

3. Evaluation and Analysis

3.1. Experimental Setup

3.1.1. Dataset

We evaluated the proposed RISE framework on a benchmark dataset WSJ0-2mix[26]. It consists of 30 hours of training set comprised of 20000 utterances from $G = 101$ speakers, 10 hours of validation set comprised of 5000 utterances from the same 101 speakers, and 5 hours of test data comprised of 3000 utterances from 18 speakers unseen during training.

3.1.2. Model setup

As mentioned before, for comparison purposes, the pre-training model in RISE was chosen to have the same basic block and model size as those of DPRNN[12], i.e., $B + B' = 6$ consecutive DPRNN blocks, of which $B = 4$ blocks were used for the Generic Extractor. The remaining $B' = 2$ blocks were used for the Separator. Likewise, in fine-tuning, $B1 = 2$ blocks were used for the Identifier, and $B2 = 2$ blocks were used for the Separator.

The deep feature dimension was set as $N = 64$; the chunk size was set as $K = 64$. We empirically set the hyperparameters as follows: $\gamma = 3, \lambda = 10, \varepsilon = 0.05$, and the learning rate μ with an initial value of 0.001 and a decaying rate of 0.96 for every two epochs using an Adam optimizer.

For each training epoch, each clean utterance in the WSJ0-2mix training set masked with a different random utterance from the same training set at random starting positions, and the SIR value was randomly sampled from a uniform distribution of 0 to 5dB. A training process was considered converged if no lower validation loss appeared in 10 consecutive epochs. For testing, we used the same pre-mixed set as [26].

3.2. Results and Discussion

3.2.1. Generalizability study

We compared the generalizability of the learnt speaker embeddings by the three schemes. For visualization, we projected the speaker embeddings $\bar{\mathbf{\Gamma}}$ to a 3-D space via PCA in Fig. 2, where each dot denotes an embedding vector extracted from an utterance, and the dots has the same color if from the same speaker: On the left, it shows embeddings of 8 random speakers from the training set (since all the three schemes gave similar plots in training, we only displayed the one using scheme #3); in the mid-left, it shows embeddings of 8 ramdom test speakers, who are different than those seen during training, and their embeddings were computed with model using scheme #1; on the mid-right, it shows the same 8 random unseen speaker, and their embeddings were computed with the model using scheme #3.

Although discriminative embeddings could be well learnt for those seen speakers as shown on the first plot in Fig.2, the discriminative power could hardly hold for zero-shot speakers than seen during training by the model using scheme #1, as shown in the mid-left (the scheme #2 also produced similarly poor discriminability, so we neglected its plot to save space); in contrast, the scheme #3 gives embeddings with a substantial discriminative property. We analyzed the key factor of the scheme #3 being different from the scheme #1 and #2. It turns out the model is purged from learning a trivial task for predicting the speaker identity information; instead, it enforces the model to learn the representations more towards a self-supervised way.

We observed all the schemes converged to comparable $\mathcal{L}_{\text{ID}}$ during training, but the scheme #1 and #2 could converge much faster than #3. This could be a suitable property for some training scenarios such as NLP tasks with enumerable tokens given a predefined fixed dictionary or standard speaker identification tasks assuming a vast number of training speakers. Arguably, these models "memorized" the token/ speaker identities rather than learning deep representations with essential discriminative power and generalization ability. In contrast, if the model has relatively little training data or if the identities in test scenarios are inevitably agnostic, we may want to avoid such a "memorization" process during learning, i.e., to prohibit the model from ending up learning the easier identity-prediction task.

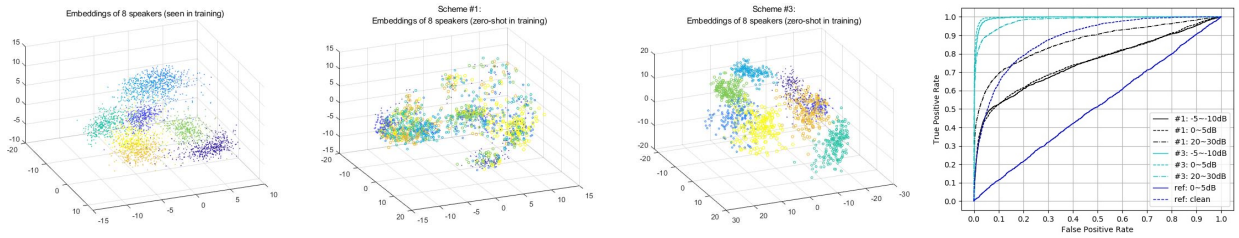

Figure 2: *3-D PCA of the speaker embeddings: (left) from 8 random speakers seen in training, (mid-left) from 8 random test speakers with zero-shot in training, following scheme #1, (mid-right) from the same 8 test speakers with zero-shot in training, following scheme #3, and (right) ROC curves of speaker verification by different models under various SIR conditions.*

3.2.2. Speaker verification performance

The purpose of this experiment was to evaluate the discriminative power of the learnt representations for zero-shot speakers. We conducted an ad-hoc down-stream task of speaker verification (SV) for this purpose. First, during enrollment, 20 utterances of each speaker were randomly drawn from the test set and composed a WSJ0-enrollment set; the speaker embeddings $\widehat{\mathbf{\Gamma}}$ extracted in RISE were collected, normalized, and averaged to generate one target vector per speaker. Then, during an evaluation, the speaker embeddings $\widehat{\mathbf{\Gamma}}$ extracted by RISE with the remaining utterances (WSJ0-test) were collected, normalized, and measured with their cosine distance to the target vectors of all speakers. We used EER and AUC as metrics to indicate the discriminability of the learnt representations.

As shown in the last plot in Fig.2, Operating Characteristic Curves (ROC) by three models under different SIR were plotted for comparison. The SIR conditions of each ROC marked on the right bottom in the last plot of Fig.2. The RISE model uses scheme #1 and #3. A conventional dvector-based SV model [27] as reference was trained on the clean training set of WSJ0-2mix, following the setup and SAD pipeline in [27]. We reproduced EER (0.21) and AUC (0.88) scores on clean WSJ0-enrollment and WSJ0-test, similar to that reported in [27] on Voxceleb. Currently we are implementing another more advanced syncnet-based SV model [28], which gives much better EER (0.024) on the clean data. Still, our EER (0.023) on $0 \sim 5$dB interfering condition has already been comparable to (or even slightly better than) its score on clean data. In future work we will choose [28] as a more substantial reference.

The AUC of the reference system in $0 \sim 5$dB SIR was close to 0.5, indicating this conventional SV system thoroughly failed in adverse interference. In contrast, despite in SIR ($-5 \sim -10$dB) worse than the SIR range during training ($0 \sim 5$dB), RISE using scheme #3 could still obtain excellent EER (0.033) and AUC (0.99) scores. Slightly worse scores of EER (0.08) and AUC (0.98) were obtained in the $20 \sim 30$dB condition, mainly due to a larger discrepancy between training and testing SIR ranges. Still, these scores indicated substantial generalizability and the discriminability of the learnt representations. RISE scheme #3 suggests it is feasible to learn better representations in adverse interfering conditions, and be free from requiring clean conditions plus a long pre-processing pipeline, including SAD, segmentation, and overlap detector, etc., as conventional SV systems would generally require. As we speculated in Sec.3.2.1, the superiority of scheme #3 over scheme #1 can be verified by comparing their ROCs.

3.2.3. Separation performance

We then compared our proposed RISE system on speech separation performance to the state-of-the-art DPRNN model [12]. Note that performances for both systems could be consistently improved by further decreasing the hyper-parameter window size (filter length) in the encoder and decoder, but at the cost of proportionately increased training time, which had not been indicated in [12]. Since our work in this paper is independent and different than optimization regarding time resolution, we set the window size to 16 samples (otherwise too large memory consumption for large-scale industrial datasets.) We evaluated the two systems under the same setup for a fair comparison.

The RISE system here used scheme #3 and was evaluated under the two inference modes as described in Sec.2.2.3. All systems were assessed in terms of SI-SNRi[23]. As shown in table 1, both the "Guided" and the "Online" modes gave superior SI-SNRi over DPRNN.

In the "online" mode, RISE performed both the **Identifier** and **Separator** tasks simultaneously. Alternatively, to reduce the inference complexity, the inference mode could be switched from the "online" to the "guided" mode after sufficient speaker embeddings had been accumulated during the "online" mode. The delay before switching could be less than 40s, which is applicable to most industrial applications. Note that the "guided" mode uses enrollments collected from the same adverse SIR environment as "online" inference, i.e., the enrollment process in RISE is very straightforward and can be performed online. Consequently, the online enrollment process can be made transparent to users.

Table 1: *Separation performances on WSJ0-2mix by the DPRNN system and our proposed RISE in different modes.*

System	Mode	#Param.	val. SI-SNRi (dB)	SI-SNRi (dB)
DPRNN	-	2.6M	16.5	15.9
RISE	Guided	2.6M	**17.7**	**16.4**
	Online	2.6M+1M (for ID task)	**17.9**	**17.5**

4. Conclusions

We proposed a novel framework called RISE, which performed compositional learning for previously independent speech separation and speaker identification tasks. The proposed pre-training and fine-tuning procedure allowed for high training efficiency. We investigated different representation learning schemes in RISE and suggested an argument about NOT memorizing the identities during training, so that the model could be enforced to learn substantially more generalizable and discriminative representations. Experiment results on down-stream tasks showed that our learnt representations had superior discriminative power than a traditional SV method. Moreover, RISE achieved consistently higher SI-SNRi in different inference mode over a state-of-the-art speech separation system.

5. References

[1] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSP)*, 2018, p. 5329–5333.

[2] J. L. Yi Liu, Liang He, "Large margin softmax loss for speaker verification," in *Proc. INTERSPEECH*, 2019.

[3] Y. B. Mirco Ravanelli, "Speaker recognition from raw waveform with sincnet," in *Proceedings of SLT 2018*, 2018.

[4] Y. Fujita, N. Kanda, S. Horiguchi, K. Nagamatsu, and S. Watanabe, "End-to-end neural speaker diarization with permutation-free objectives," in *Proc. INTERSPEECH*, 2019.

[5] Z. Huang, S. Watanabe, Y. Fujita, P. Garcia, Y. Shao, D. Povey, and S. Khudanpur, "Speaker diarization with region proposal network," in *IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSP)*, 2020.

[6] E. C. Cherry, "Some experiments on the recognition of speech, with one and with two ears," *The Journal of the acoustical society of America*, vol. 25, no. 5, pp. 975–979, 1953.

[7] Y. Luo and N. Mesgarani, "Tasnet: time-domain audio separation network for real-time, single-channel speech separation," in *IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSP)*. IEEE, 2018, pp. 696–700.

[8] D. Yu, M. Kolbæk, Z.-H. Tan, and J. Jensen, "Permutation invariant training of deep models for speaker-independent multi-talker speech separation," in *IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSP)*. IEEE, 2017, pp. 241–245.

[9] M. Kolbæk, D. Yu, Z.-H. Tan, J. Jensen, M. Kolbaek, D. Yu, Z.-H. Tan, and J. Jensen, "Multitalker speech separation with utterance-level permutation invariant training of deep recurrent neural networks," *TASLP*, vol. 25, no. 10, pp. 1901–1913, 2017.

[10] S. Bai, J. Z. Kolter, and V. Koltun, "An empirical evaluation of generic convolutional and recurrent networks for sequence modeling," *arXiv preprint arXiv:1803.01271*, 2018.

[11] 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.

[12] Y. Luo, Z. Chen, and T. Yoshioka, "Dual-path rnn: efficient long sequence modeling for time-domain single-channel speech separation," *arXiv preprint arXiv:1910.06379*, 2019.

[13] N. Zeghidour and D. Grangier, "Wavesplit: End-to-end speech separation by speaker clustering," in *arXiv preprint arXiv:2002.08933*, 2020.

[14] M. Pariente, S. Cornell, J. Cosentino, S. Sivasankaran, E. Tzinis, J. Heitkaemper, M. Olvera, F. Stoter, M. Hu, J. M. M. Donas, D. Ditter, A. Frank, A. Deleforge, and E. Vincent, "Asteroid: the pytorch-based audio source separation toolkit for researchers," *arXiv preprint arXiv:2005.04132*, 2020.

[15] R. Getzmann, S.; Naatanen, "The mismatch negativity as a measure of auditory stream segregation in a simulated "cocktail-party" scenario: effect of age," *Neurobiology of Age*, 2015.

[16] R. Gerrig and P. Zimbardo, *Psychology and Life*. Allyn and Bacon, 2011.

[17] S. Pascual, M. Ravanelli, J. Serra, A. Bonafonte, and Y. Bengio, "Learning problem-agnostic speech representations from multiple self-supervised tasks," in *Proc. INTERSPEECH*, 2019.

[18] M. Ravanelli and Y. Bengio, "Learning speaker representations with mutual information," in *Proc. INTERSPEECH*, 2019.

[19] J. Chorowski, R. J. Weiss, S. Bengio, and A. Oor, "Unsupervised speech representation learning using wavenet autoencoders," in *IEEE TASLP*, 2019.

[20] M. Ravanelli, J. Zhong, S. Pascual, P. Swietojanski, J. Monteiro, J. Trmal, and Y. Bengio, "Multi-task self-supervised learning for robust speech recognition," in *IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSP)*, 2020.

[21] W. L. Taylor, "Cloze procedure: A new tool for measuring readability," *Journalism Bulletin*, vol. 30, no. 4, pp. 415–433, 1953.

[22] K. L. Jacob Devlin, Ming-Wei Chang and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics*, 2019, pp. 4171–4186.

[23] J. Le Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, "Sdr–half-baked or well done?" in *2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 626–630.

[24] D. A. Hudson and C. D. Manning, "Compositional attention networks for machine reasoning," in *Proceedings of the International Conference on Learning Representations (ICLR)*, 2018.

[25] E. Perez, F. Strub, H. Vries, V. Dumoulin, and A. Courville, "Film: Visual reasoning with a general conditioning layer," in *arXiv preprint arXiv:1709.07871*, 2017.

[26] J. R. Hershey, Z. Chen, J. Le Roux, and S. Watanabe, "Deep clustering: Discriminative embeddings for segmentation and separation," in *IEEE International Conference on Acoustics, Speech and Signal Processing(ICASSP)*. IEEE, 2016, pp. 31–35.

[27] A. Torfi, J. Dawson, and N. M. Nasrabadi, "Text-independent speaker verification using 3d convolutional neural networks," in *2018 IEEE International Conference on Multimedia and Expo (ICME)*, 2018.

[28] M. Ravanelli and Y. Bengio, "Speaker recognition from raw waveform with sincnet," in *In Proceedings of SLT 2018*, 2018.