LANGUAGE-QUERIED AUDIO SOURCE SEPARATION VIA RESUNET WITH DPRNN

Technical Report

Han Yin[1], *Jisheng Bai*[1,3,4], *Mou Wang*[2], *Jianfeng Chen*[1]

[1] Joint Laboratory of Environmental Sound Sensing, School of Marine Science and Technology, Northwestern Polytechnical University, Xi'an, China
[2] Institute of Acoustics, Chinese Academy of Sciences, Beijing, China
[3] School of Electrical & Electronic Engineering, Nanyang Technological University, Singapore
[4] LianFeng Acoustic Technologies Co., Ltd. Xi'an China

ABSTRACT

This report presents our submitted systems for the task 9 of DCASE challenge: language-queried audio source separation (LASS). LASS is the task of separating arbitrary sound sources using textual descriptions of the desired source, also known as "separate what you describe". Specifically, we first incorporate a dual-path recurrent neural network (DPRNN) block into ResUNet, which is significantly beneficial for improving the separation performance. Then, we trained the proposed model using a large number of public datasets, including Clotho, FSD50K, Audiocaps, Auto-ACD, and Wavcaps. We trained the proposed model at 16 kHz and 32 kHz respectively, and the 32 kHz model achieved the best separation performance with an SDR of 8.191 dB on the validation set, which is 2.483 dB higher than the challenge baseline.

Index Terms— Language-queried audio source separation, DPRNN, ResUNet

1. INTRODUCTION

Language-queried audio source separation (LASS) [1] is the task of separating sound sources using textual descriptions of the desired source. LASS provides a useful tool for future source separation systems, allowing users to extract audio sources via natural language instructions. Such a system could be useful in many applications, such as automatic audio editing, multimedia content retrieval, and augmented listening. The objective of LASS is to effectively separate sound sources using natural language queries, thereby advancing the way we interact with and manipulate audio content.

As shown in Fig. 1, a LASS system is composed of a query encoder and an audio source separation model. The query encoder is used to convert textual descriptions into embeddings, which are incorporated into the separation model. The challenge baseline system [1] uses the text encoder of contrastive language-audio pretraining model (CLAP) [2] as the query encoder, and applies ResUNet [3] as the separation model.

ResUNet is a time-frequency domain model, consisting of an encoder and a decoder. In ResUNet, convolutional layers are used to extract local invariant features from the spectrogram input. However, convolutional layers cannot effectively extract dependencies between frequency bins or temporal frames. Therefore, we incorporate dual-path recurrent neural network (DPRNN) [4] into ResUNet to learn dynamic audio information. DPRNN is composed of two bidirectional-LSTMs (Bi-LSTMs) [5], which can be used to extract acoustic information in the frequency domain and temproal domain iteratively.

[1] https://dcase.community/challenge2024/task-language-queried-audio-source-separation

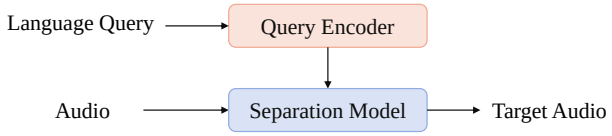

Figure 1: The architecture of LASS system.

2. METHODS

2.1. ResUNet with DPRNN

In order to perform dynamic feature extraction, we incorporate a DPRNN block between the encoder and decoder in ResUNet to iteratively apply time-domain and frequency-domain modeling. Fig. 2 shows the architecture of ResUNet and ResUNet with DPRNN.

As shown in Fig. 3, the DPRNN block is mainly composed of two Bi-LSTMs. We pass the embedding $\mathbf{Z} \in \mathbb{R}^{C \times T \times F}$ generated by the encoder through Bi-LSTMs in the time domain and frequency domain sequentially, where C, T, and F are the number of channels, frames, and frequencies, respectively. In this way, the DPRNN block can effectively extract dependencies between frames and frequency bins, improving the model's semantic understanding of dynamic audio information.

2.2. Loss function

We train the model end-to-end using an L1 loss function between the predicted and target waveforms. Since waveform-based L1 loss is simple to implement and has shown good performance on universal sound separation tasks [6].

$$Loss = ||s - \hat{s}||_1 \tag{1}$$

where s is the target waveform and $\hat{s}$ is the separated waveform. The lower loss value indicates that the separated signal $\hat{s}$ is closer to the ground truth signal s.

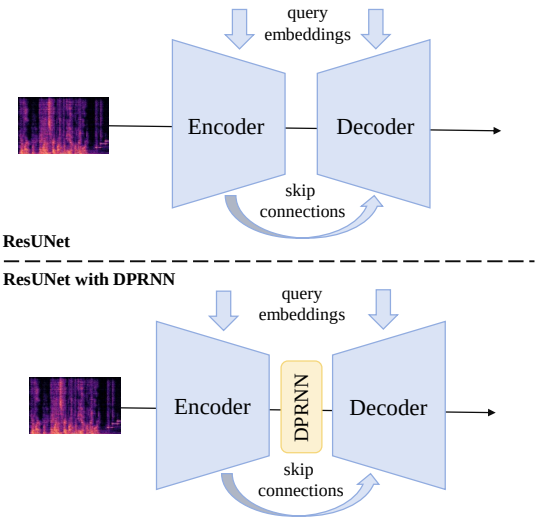

Figure 2: The architecture of ResUNet/ResUNet with DPRNN.

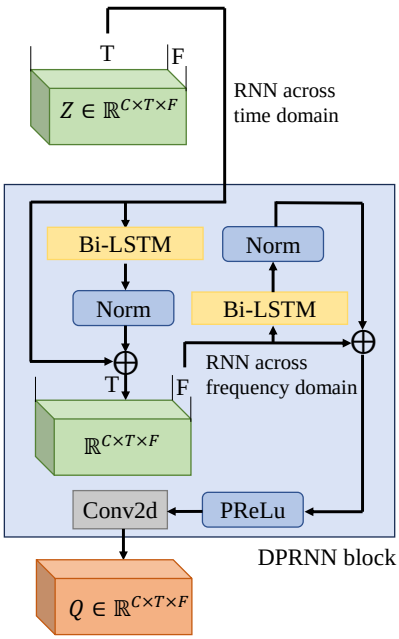

Figure 3: The architecture of the DPRNN block.

3. EXPERIMENTAL SETUPS

3.1. Datasets

In this task, publicly available audio-caption paired datasets are used for training and validation. For training, we use data from the following datasets:

⋆ DCASE challenge task 9 development dataset in 2024 (DCASE-T9Dev-2024)[1]: This dataset is composed of audio samples from FSD50K [7] and Clotho v2 [8] datasets. Clotho v2 consists of 6972 audio samples, each audio clip is labeled with five captions. And FSD50K contains over 51K audio clips manually labeled using 200 classes drawn from the AudioSet [9] ontology. For each audio clip in the FSD50K dataset, one automatic caption was generated for each audio clip by prompting GPT-4 [10] with its sound event tags.

⋆ Audiocaps [11]: This is a large-scale dataset of 46K audio clips with human-written text pairs collected via crowdsourcing on the AudioSet dataset.

⋆ Auto-ACD [12]: This dataset contains over 1.9M audio-text pairs, which were generated based on a series of public tools or APIs, in which audio samples were from AudioSet and VGGSound [13]. We selected 100K audio clips from the Audioset portion of Auto-ACD for training.

⋆ Wavcaps [14]: This dataset is the first large-scale weakly-labelled audio captioning dataset, comprising approximately 400K audio clips with paired captions. Audio clips and their raw descriptions were from diverse sources, including FreeSound[2], BBC Sound Effects[3], SoundBible[4], and AudioSet. We sampled 200K audio clips from FreeSound portion of Wavcaps for training.

For validation, we use data from DCASE challenge task 9 validation dataset in 2024 (DCASE-T9Val-2024)[1]. The dataset contains 2100 audio clips, which were sampled from data uploaded to FreeSound between April and October 2023. Each audio file has been chunked into a 10-second clip and converted to mono 16 kHz.

[2]https://dcase.community/challenge2021
[3]https://sound-effects.bbcrewind.co.uk/
[4]https://soundbible.com/

3.2. Evaluation Metrics

We use SDR (signal-to-distortion ratio), SDRI (signal-to-distortion ratio improvement) and SI-SDR (scale-invariant signal-to-distortion ratio) for evaluation.

3.3. Training Setups

For 16 kHz models, we resample all training audio samples to 16 kHz and convert the waveform to spectrogram using short-time Fourier transform (STFT) with a window size of 1024 and a hop size of 160. Similarly, for 32 kHz models, we resample training audio clips at 32 kHz, then get their spectograms through STFT with a window size of 2048 and a hop size of 320. We apply an Adam optimizer with a learning rate of 0.0001 to train the model with the batch size of 12 on 2 NVIDIA RTX 3090 GPU cards.

It should be noted that because audio samples in DCASE-T9Val-2024 are 16kHz, when we evaluate the 32kHz model, the audio is first upsampled to 32kHz and fed into the model, and then the output is downsampled to 16kHz.

4. RESULTS AND DISCUSSIONS

Then challenge baseline system[1] trained ResUNet-16k on DCASE-T9Dev-2024 for 200K steps. In order to explore the effectiveness of the DPRNN block, following the baseline, we trained ResUNet-16k with DPRNN using the same settings. Table 3 shows the performance of ResUNet-16k with/without DPRNN on DCASE-T9Val-2024. By incorporating the DPRNN block into ResUNet-16k, the separation performance is improved, with an improvement of 0.27 dB on SDR, which demonstrates that the Bi-LSTMs in DPRNN play an important role in audio separation process.

Then, we use all training data to finetune the models for better separation performance. For 16 kHz models, we use the challenge baseline checkpoint[1] as the initialization parameter, and for

Table 1: Results of ResUNet-16k with/without DPRNN on DCASE-T9Val-2024.

Model	DPRNN	SDR ↑	SDRI ↑	SI-SDR ↑
ResUNet-16k[1]	✗	5.708 dB	5.673 dB	3.862 dB
	✓	**5.978** dB	**5.943** dB	**3.975** dB

32 kHz models, we use ResUNet-32k [15] as the initialization parameter. Table 2 presents the results on DCASE-T9Val-2024, it can be seen that in both scenarios (16 kHz and 32 kHz), the model with DPRNN can achieve better performance than the model without DPRNN. And the improvement brought by DPRNN on ResUNet-32k is smaller than that on ResUNet-16k. One potential reason may be that we need more steps to obtain better performance of DPRNN.

Table 2: Results of ResUNet-16k and ResUNet-32k with/without DPRNN on DCASE-T9Val-2024.

Model	DPRNN	SDR ↑	SDRI ↑	SI-SDR ↑
Init				
ResUNet-16k[1]	✗	5.708 dB	5.673 dB	3.862 dB
Finetune (2M steps)				
ResUNet-16k	✗	7.087 dB	7.052 dB	5.413 dB
ResUNet-16k	✓	8.007 dB	7.972 dB	6.459 dB
Init				
ResUNet-32k [15]	✗	8.009 dB	7.974 dB	6.533 dB
Finetune (1M steps)				
ResUNet-32k	✗	8.047 dB	8.012 dB	6.558 dB
ResUNet-32k	✓	**8.191** dB	**8.156** dB	**6.794** dB

We finally submitted 4 systems to the chanllenge, details are described as follows:

- submission 1: ResUNet-16k without DPRNN (single model).
- submission 2: ResUNet-16k with DPRNN (single model).
- submission 3: ResUNet-32k with DPRNN (single model).
- submission 4: ResUNet-32k with DPRNN + ResUNet-32k without DPRNN + ResUNet-16k with DPRNN (ensemble).

5. CONCLUSIONS

This technical report outlines our research on DCASE Challenge Task 9, focusing on language-queried audio source separation. Initially, we integrated the DPRNN block into ResUNet, demonstrating its advantageous impact on enhancing source separation performance. Subsequently, through the utilization of extensive public datasets for model fine-tuning, we have achieved improved separation results. Notably, our proposed ResUNet-32k with DPRNN has shown considerable advancement over the baseline, exhibiting an SDR of 8.191 dB on DCASE-T9Val-2024. By ensembling 32kHz models and the 16kHz model, the separation performance can be further improved, with an SDR of 8.467 dB.

Table 3: Results of submitted systems on DCASE-T9Val-2024.

Model	SDR ↑	SDRI ↑	SI-SDR ↑
Submission 1	7.087 dB	7.052 dB	5.413 dB
Submission 2	8.007 dB	7.972 dB	6.459 dB
Submission 3	8.191 dB	8.156 dB	6.794 dB
Submission 4	**8.467** dB	**8.432** dB	**7.403** dB

6. REFERENCES

[1] X. Liu, H. Liu, Q. Kong, X. Mei, J. Zhao, Q. Huang, M. D. Plumbley, and W. Wang, "Separate what you describe: Language-queried audio source separation," *arXiv preprint arXiv:2203.15147*, 2022.

[2] Y. Wu, K. Chen, T. Zhang, Y. Hui, T. Berg-Kirkpatrick, and S. Dubnov, "Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation," in *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2023, pp. 1–5.

[3] Q. Kong, Y. Cao, H. Liu, K. Choi, and Y. Wang, "Decoupling magnitude and phase estimation with deep resunet for music source separation," *arXiv preprint arXiv:2109.05418*, 2021.

[4] Y. Luo, Z. Chen, and T. Yoshioka, "Dual-path rnn: efficient long sequence modeling for time-domain single-channel speech separation," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 46–50.

[5] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[6] Q. Kong, K. Chen, H. Liu, X. Du, T. Berg-Kirkpatrick, S. Dubnov, and M. D. Plumbley, "Universal source separation with weakly labelled data," *arXiv preprint arXiv:2305.07447*, 2023.

[7] E. Fonseca, X. Favory, J. Pons, F. Font, and X. Serra, "Fsd50k: an open dataset of human-labeled sound events," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 829–852, 2021.

[8] K. Drossos, S. Lipping, and T. Virtanen, "Clotho: An audio captioning dataset," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 736–740.

[9] J. F. Gemmeke, D. P. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "Audio set: An ontology and human-labeled dataset for audio events," in *2017 IEEE international conference on acoustics, speech and signal processing (ICASSP)*. IEEE, 2017, pp. 776–780.

[10] J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, *et al.*, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.

[11] C. D. Kim, B. Kim, H. Lee, and G. Kim, "Audiocaps: Generating captions for audios in the wild," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, 2019, pp. 119–132.

[12] L. Sun, X. Xu, M. Wu, and W. Xie, "A large-scale dataset for audio-language representation learning," *arXiv preprint arXiv:2309.11500*, 2023.

[13] H. Chen, W. Xie, A. Vedaldi, and A. Zisserman, "Vggsound: A large-scale audio-visual dataset," in *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2020, pp. 721–725.

[14] X. Mei, C. Meng, H. Liu, Q. Kong, T. Ko, C. Zhao, M. D. Plumbley, Y. Zou, and W. Wang, "Wavcaps: A chatgpt-assisted weakly-labelled audio captioning dataset for audio-language multimodal research," *arXiv preprint arXiv:2303.17395*, 2023.

[15] X. Liu, Q. Kong, Y. Zhao, H. Liu, Y. Yuan, Y. Liu, R. Xia, Y. Wang, M. D. Plumbley, and W. Wang, "Separate anything you describe," *arXiv preprint arXiv:2308.05037*, 2023.