

# Separate What You Describe: Language-Queried Audio Source Separation

Xubo Liu<sup>1</sup>, Haohe Liu<sup>1</sup>, Qiuqiang Kong<sup>2</sup>, Xinhao Mei<sup>1</sup>, Jinzheng Zhao<sup>1</sup>,  
Qiushi Huang<sup>1</sup>, Mark D. Plumbley<sup>1</sup>, Wenwu Wang<sup>1</sup>

<sup>1</sup>School of Computer Science and Electronic Engineering, University of Surrey, UK

<sup>2</sup>Speech, Audio, and Music Intelligence (SAMI) Group, ByteDance, China

{xubo.liu, hl01486, m.plumbley, w.wang}@surrey.ac.uk, kongqiuqiang@bytedance.com

## Abstract

In this paper, we introduce the task of language-queried audio source separation (LASS), which aims to separate a target source from an audio mixture based on a natural language query of the target source (e.g., “*a man tells a joke followed by people laughing*”). A unique challenge in LASS is associated with the complexity of natural language description and its relation with the audio sources. To address this issue, we proposed LASS-Net, an end-to-end neural network that is learned to jointly process acoustic and linguistic information, and separate the target source that is consistent with the language query from an audio mixture. We evaluate the performance of our proposed system with a dataset created from the AudioCaps dataset. Experimental results show that LASS-Net achieves considerable improvements over baseline methods. Furthermore, we observe that LASS-Net achieves promising generalization results when using diverse human-annotated descriptions as queries, indicating its potential use in real-world scenarios. The separated audio samples and source code are available at <https://liuxubo717.github.io/LASS-demopage>.

**Index Terms:** universal sound separation, source separation, target source separation, natural language processing

## 1. Introduction

Human beings can focus their auditory attention on specific sounds in environments [1]. Source separation systems aim to separate mixtures of sound sources, which is the basis for computational auditory scene analysis [2]. Recently, significant progress has been made in audio source separation such as speech separation [3, 4], music source separation [5, 6, 7] and universal sound separation [8, 9, 10, 11].

One major challenge for existing source separation systems is to deal with a vast number of sound classes in the real world. When multiple sources are presented simultaneously, it is difficult for these systems to obtain accurate separation results [12, 13]. Previous methods [8, 13] investigated using source category information as a query to separate a specific source from a mixture, as a way to reduce the difficulty of source separation. However, the information provided by source categories is often limited. In practical applications, instead of using a predefined fixed set of source categories, one may prefer to use a natural language description to identify and separate the target sound source. Such natural language descriptions can include auxiliary information for source separation such as spatial and temporal relationships of sound events, such as “*dog barks in the background*” or “*people applaud followed by a woman speaking*”. To our knowledge, audio source separation with natural language queries has not been investigated in the literature.

This paper introduces a language-queried audio source separation (LASS) task. Given an audio mixture and a natural

language query of the target source, LASS aims to automatically extract the target source from the mixture, where the target source is consistent with the language query. LASS provides a potentially useful tool for future source separation systems, allowing users to extract desired audio sources via natural language instructions. Such a system could be useful in many applications, such as automatic audio editing [14], multimedia content retrieval [15], and controllable hearable devices [13].

The challenges of achieving LASS are associated with the complexity of natural language expressions and the characterization of their relation with sound sources. The language description of a sound source usually consists of multiple phrases (e.g., “*people speak and music plays*”), each phrase referring to a sound event in the audio mixture. In addition, the same audio source can be delivered with diverse language expressions, such as “*music is being played with a rhythmic beat*” or “*an upbeat music melody is playing over and over again*”. In summary, LASS not only requires these phrases and their relationships to be captured in the language description, but also one or more sound sources that match the language query should be separated from the audio mixture.

In this work, we present LASS-Net, which is trained to jointly process acoustic and linguistic information, and separate the target source described by the natural language expressions. In LASS-Net, a Transformer-based [16] query network is used to encode the language expression into a query embedding, and a ResUNet-based [5] separation network is then used to separate the target source from mixture conditioned on the query embedding. To evaluate the performance of our model, we create a dataset based on the AudioCaps [17] dataset, experimental results demonstrate that our model can achieve considerable improvements over baseline methods. We also observe that LASS-Net shows promising separation performance when queried by diverse human-annotated descriptions, indicating the potential for generalization in real application scenarios.

The remainder of this paper is organized as follows. We review the related work in Section 2. The LASS-Net is presented in Section 3. We describe the dataset we created in Section 4, and experiments and results in 5. We draw our conclusions in Section 6, together with discussions on future work.

## 2. Related Work

Our work relates to several tasks in the literature: universal sound separation, target source extraction, and audio captioning. We will discuss each of these as follows.

### 2.1. Universal sound separation

Universal sound separation (USS) [9] is the task of separating a mixture of arbitrary sound sources in terms of their classes. USS is a challenging problem since the number of sound classes in the

e.g., People speak and dog barks

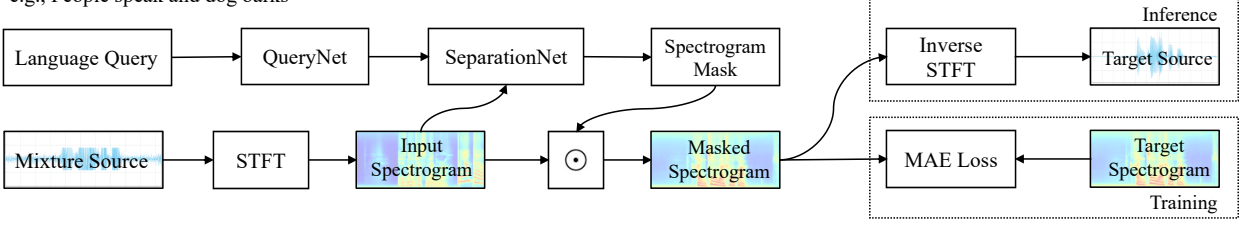


Figure 1: Framework of our proposed LASS-Net.

real world is very large. Several approaches have been proposed to address the issue of a large number of sound classes, such as leveraging semantic information learned by sound classifiers [12] and establishing large-scale high-quality datasets [11]. In a similar way to USS, LASS aims to separate real-world sounds, but the objective of LASS is to perform the separation using natural language descriptions as queries.

## 2.2. Target source extraction

Target source extraction (TSE) aims to separate a specific source from an audio mixture given query information such as a sound event tag. In contrast to USS, TSE only extracts sources of interest from the mixture. There are several applications for this problem, such as target speech extraction using speaker information [18, 19], and target sound extraction using acoustic event tags [13] or onomatopoeic words [20]. In contrast to TSE, LASS focuses on extracting target audio sources that match linguistic queries.

## 2.3. Automated audio captioning

Automated Audio captioning (AAC) [21, 22, 23, 24] is the task where natural language descriptions are generated for an audio clip. Recently, AAC has attracted increasing interest in the Detection and Classification of Acoustic Scenes and Events (DCASE) community. There are several datasets for AAC, such as Clotho [25] and AudioCaps [17]. AudioCaps is the largest public dataset for AAC research and has been used in several recent studies [26, 27]. In this work, we create a dataset for LASS based on the AudioCaps dataset. This new dataset will be described in Section 4.

# 3. Proposed Approach

We propose LASS-Net, a neural network which separates target audio sources with natural language queries, as shown in Figure 1. LASS-Net consists of two components: a query network (QueryNet) that takes a language query as input and outputs a query embedding, and a separation network (SeparationNet) that takes mixture and query embedding as input and predicts the target source. These two modules are trained jointly. We describe the details of LASS-Net in the following sections.

## 3.1. LASS-Net

In LASS-Net, We build the separation in the time-frequency domain. We denote the language query as  $q$ , and deploy a query network to extract a query embedding  $e_q$ :

$$\text{QueryNet}(q) \mapsto e_q. \quad (1)$$

The audio mixture  $x$  is transformed to the spectrogram  $X$  using short-time Fourier transform (STFT). The magnitude spectrogram and phase of  $X$  are denoted as  $|X|$  and  $e^{j\angle X}$ , where  $X = |X|e^{j\angle X}$ . Magnitude spectrogram  $|X| \in \mathbb{R}^{F \times T}$  is a two-dimensional time-frequency feature, where  $T$  and  $F$  represent the number of time frames and the dimension of the spectral feature, respectively. Our objective is to learn a regression from  $|X|$  to the magnitude spectrogram  $|\hat{S}|$  of the estimated target source  $\hat{s}$ , conditioned on the query embedding  $e_q$ . Specifically, we use a separation network that accepts inputs  $|X|$  and  $e_q$ , and outputs a latent feature  $Z$  with the same shape as  $|X|$ :

$$\text{SeparationNet}(|X|, e_q) \mapsto Z \in \mathbb{R}^{F \times T}. \quad (2)$$

Then,  $Z$  is fed into an element-wise sigmoid function  $\sigma(\cdot)$  to obtain a spectrogram mask  $M$ :

$$\sigma(Z) = M \in [0, 1]^{F \times T}. \quad (3)$$

The magnitude spectrogram  $|\hat{S}|$  of the estimated target source is obtained by masking  $|X|$ :

$$|\hat{S}| = M \odot |X| \in \mathbb{R}^{F \times T}, \quad (4)$$

where  $\odot$  is the Hadamard product.

The training objective is to minimize the mean absolute error (MAE) between  $|\hat{S}|$  and magnitude spectrogram  $|S| \in \mathbb{R}^{F \times T}$  of the ground truth target audio source. The MAE loss term is defined as follows:

$$\text{Loss}_{\text{MAE}} = \left\| |S| - |\hat{S}| \right\|_1, \quad (5)$$

where  $\|\cdot\|$  is an  $l_1$  norm.

The mixture phase  $e^{j\angle X}$  is reused to recover the STFT spectrogram  $\hat{S}$  from the estimated magnitude spectrogram  $|\hat{S}|$ , where  $\hat{S} = |\hat{S}|e^{j\angle X}$ . Finally, inverse STFT is applied on  $\hat{S}$  to obtain the estimated source  $\hat{s}$ .

## 3.2. Query network

To extract the language query embedding, we use BERT [16] as a query network in the proposed LASS-Net. BERT is a language model pre-trained on large-scale text datasets (e.g., BooksCorpus [28]). BERT contains prior linguistic knowledge such as syntactic and semantic information, which is useful in audio-language tasks, as shown in audio captioning [27, 29]. Concretely, we use the pre-trained BERT [30] consisting of 4 Transformer encoder blocks, each with 4 head and 256 hidden dimensions, respectively. The input language query  $q = \{q_n\}_{n=1}^N$  which consists of  $N$  words is fed into the BERT model, as a result, a 256-dimensional word-level embedding  $e = \{e_n\}_{n=1}^N$  is obtained. In the BERT model, the bidirectional self-attention is used to

consider both past and future context, therefore, we adopt the first embedding  $e_1$  as the output embedding of BERT. Finally,  $e_1$  is passed to a fully-connected layer with 256 nodes and ReLU activation to obtain the language query embedding  $e_q$ .

### 3.3. Separation network

We design the separation network based on ResUNet [5], which is an improved UNet model used previously in [5, 6, 8] for source separation. The ResUNet consists of six encoder blocks and six decoder blocks. There are skip connections between encoder and decoder blocks at the same level. The encoder and decoder blocks share the same structure that contains two ConvBlocks. Each ConvBlock consists of a batch normalization, a leakyReLU activation, and a convolutional layer with kernel size  $4 \times 4$ . In the encoder blocks, average pooling is applied for downsampling. In the decoder blocks, transpose convolution is applied for upsampling. The number of feature maps of each encoder block are 32, 64, 128, 256, 384, and 384, respectively, and the number of feature maps of each decoder block is 384, 384, 256, 128, 64, and 32, respectively. After the last decoder block, a 32-channel ConvBlock followed by a  $1 \times 1$  convolutional layer is deployed to estimate the spectrogram mask, which has the same shape as the input spectrogram.

To bridge the query network and the separation network, we use the Feature-wise Linearly modulated (FiLM) layer [31] after each ConvBlock deployed in the separation network. Specifically, let  $H^{(l)} \in \mathbb{R}^{m \times h \times w}$  denote the output feature map of a ConvBlock  $l$  that has  $m$  filters. The modulation parameters are applied per feature map  $H_i^{(l)}$  with the FiLM layer as follows:

$$\text{FiLM}(H_i^{(l)} | \gamma_i^{(l)}, \beta_i^{(l)}) = \gamma_i^{(l)} H_i^{(l)} + \beta_i^{(l)}, \quad (6)$$

where  $H_i^{(l)} \in \mathbb{R}^{h \times w}$ , and  $\gamma^{(l)}, \beta^{(l)} \in \mathbb{R}^m$  are the modulation parameters from  $g(\cdot)$ , i.e.,  $(\gamma, \beta) = g(e_q)$ , such that  $g(\cdot)$  is a neural network and  $e_q$  is the language embedding obtained from the query network. In this work, we model  $g(\cdot)$  with two fully connected layers followed by ReLU activation, which is jointly trained with the separation network.

## 4. Dataset

We have created a dataset for LASS based on the AudioCaps [17] dataset. AudioCaps is the largest publicly available dataset for audio-language research, containing approximately 50k 10-second audio clips from AudioSet [32]. AudioCaps provides one human-annotated caption for each audio clip in the training set, and we can get sound event tags for each audio clip from AudioSet. Note that there is no direct connection between these two annotations. To ensure the diversity of audio clips, we first select 33 sound event tags (e.g., speech, typing, vibration, dog, rain) that belong to five root categories of AudioSet Ontology [32]: Human sounds, Animal, Sound of things, Natural sound, and Channel, environment and background. Then, we retrieve audio clips from AudioCaps if their sound event tags are all contained in these 33 sound event classes. As a result, 6244 audio clips ( $\sim 17.3$  hours) are retrieved, each audio clip has one caption in AudioCaps. We divide these audio clips into training data and test data of 6044 and 200 audio clips, respectively.

To create the audio mixtures, we first select an audio clip as the target source, and then randomly select an audio clip as the background source, whose sound event tag does not overlap with that of the target source. We mix the target source and the background source with a signal-to-noise ratio (SNR) at 0

dB. The training mixtures are created on-the-fly using training data. To create the test mixtures, each test audio is mixed with five randomly selected background sources. As a result, 1000 test mixtures are created, each mixture has a ground truth target source audio and one language query, we denote these test mixtures as LASS-Test.

In practice, human descriptions of an audio source are often diverse. To evaluate the model performance under diverse language expressions, we first randomly select 50 audio clips from 200 test audio and invite five language experts to annotate them. Each person labels one description per audio clip without any hints and restrictions. As a result, we collect an audio test subset, where each audio has six captions (one from AudioCaps and five from the annotators we recruited). Then, the corresponding 250 test audio mixtures are drawn from the LASS-Test to create a test subset mixtures, referred to LASS-Sub-Test. The details of our created dataset are made available on Github<sup>1</sup>.

## 5. Experiments and Results

### 5.1. Data processing

We load audio signals using 32 kHz sampling rate. STFT is calculated with a frame size of 1024 and a hop size of 512. An audio clip of 10 seconds results in a spectrogram with shape of  $513 \times 626$ . We convert all language descriptions to lower case and remove punctuation. One special token “<SOS>” is added to the start of each sentence. We tokenize our language descriptions corpus using the WordPiece [33] to match with the BERT pre-trained vocabulary ( $\sim 30k$ ).

### 5.2. Baseline system

Previous methods investigated separating sounds from a mixture given sound event tags [8, 13]. We implement a sound event tags-queried source separation model as a baseline system. Specifically, the baseline model is queried by a multi-hot encoding of the sound event tags of the target source in AudioSet. To make a fair comparison between the separation performance queried by language descriptions and sound event tags, the architecture we used for the baseline model is similar to that of our proposed LASS-Net, except that the query network is a fully-connected layer with 256 nodes followed by a ReLU activation.

### 5.3. Training procedure

We set batch size to 16. Adam [34] optimizer is used for training with the learning rate of  $3 \times 10^{-4}$ . We train the systems for 200 000 iterations using four Nvidia-RTX-3090-24GB GPUs.

### 5.4. Evaluation metrics

We evaluate the model performance using four commonly used objective metrics [35] in source separation: source to distortion ratio (SDR), source to inferences ratio (SIR), sources to artifact ratio (SAR) and scale-invariant SDR (SI-SDR) [36]. A higher number indicates a better performance.

Table 1: *Evaluation results on LASS-Test.*

Models	SDR	SIR	SAR	SI-SDR
Unprocessed	-0.46	12.17	0.55	-0.53
UNet-Tag	5.40	16.28	4.75	4.24
LASS-Net	5.89	16.70	5.18	4.86

<sup>1</sup><https://github.com/liuxubo717/LASS>

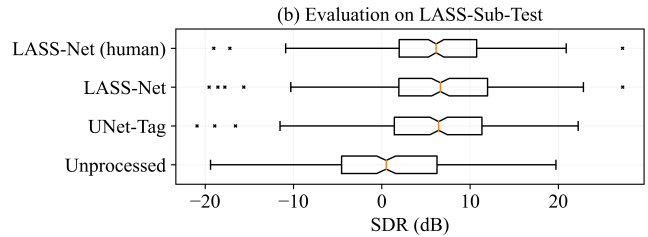
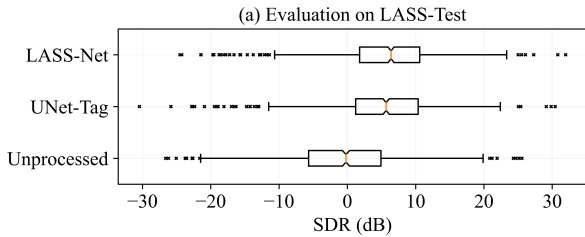


Figure 2: Box plots of the SDR scores on two experiments: (a) Evaluation on LASS-Test (b) Evaluation on LASS-Sub-Test. Red solid line represents the median value.

Table 2: Evaluation results on LASS-Sub-Test.

Models	SDR	SIR	SAR	SI-SDR
Unprocessed	0.56	12.27	1.68	0.53
UNet-Tag	6.18	16.41	5.29	4.90
LASS-Net	6.58	17.57	5.60	5.55
LASS-Net (human)	6.13	17.03	5.45	5.04

## 5.5. Results

### 5.5.1. Evaluation on LASS-Test

Three systems are evaluated on the LASS-Test in this experiment. The first method is the *Unprocessed* system which directly uses the audio mixtures in the test set for evaluation. The second one is the UNet-based baseline system we implemented using sound event tags as queries, referred to as the *UNet-Tags* system. The third one is the approach we proposed using AudioCaps language descriptions as queries, denoted as *LASS-Net*.

Table 1 shows the experimental results and Figure 2 (a) depicts the box plot of the SDR scores. *LASS-Net* achieves the SDR, SIR, SAR, SI-SDR improvement of 6.35 dB, 4.53 dB, 4.63 dB, and 5.39 dB, respectively, compared with the *unprocessed* system. This proves the feasibility of LASS task and the effectiveness of our proposed approach. In addition, compared with the *UNet-Tag* baseline system, *LASS-Net* achieves the SDR, SIR, SAR, SI-SDR improvement of 0.49 dB, 0.42 dB, 0.43 dB, and 0.62 dB, respectively. Those performance improvements indicate that using natural language descriptions as queries leads to more accurate separation results, as compared with the use of sound event tags.

### 5.5.2. Evaluation on LASS-Sub-Test

Four systems are evaluated on the LASS-Sub-Test. The first three systems are *Unprocessed*, *UNet-Tag*, and *LASS-Net*, respectively, which are same as those described in Section 5.5.1. The fourth system is the model we proposed, but using our collected human description as queries, which we denote as *LASS-Net (human)*. For *LASS-Net (human)*, we average the evaluation results of five language queries for each target audio source.

Table 2 shows the experimental results and Figure 2 (b) depicts the box plot of the SDR scores. *LASS-Net (human)* achieves the SDR, SIR, SAR, SI-SDR improvement of 5.57 dB, 4.76 dB, 3.77 dB, and 4.51 dB, respectively, compared with the *unprocessed* system. Although the human-annotated descriptions we collected have a different word distribution from that in AudioCaps, the performance of *LASS-Net (human)* is only slightly lower as compared with the *LASS-Net* system. In addition, *LASS-Net (human)* outperforms *UNet-Tag* baseline system on SIR, SAR and SI-SDR metrics by 0.62 dB, 0.16 dB, and 0.14 dB, respectively. These experimental results indicate the promising generalization performance of our proposed approach.

We visualize the spectrograms of an audio mixture, the ground truth target source, the separated sources queried by AudioCaps description and our collected human description, respectively, in Figure 3. Although these two language queries are different, we observe that the results of these two separations are similar and both are close to the ground truth source, which is consistent with our experimental results. More audio separation samples are available on our project page.

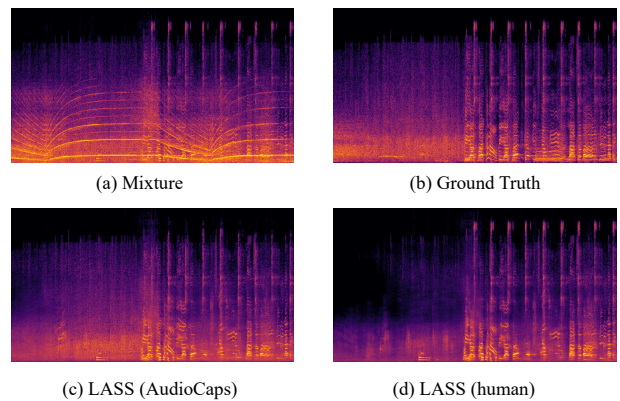


Figure 3: Spectrogram visualizations of (a) an audio mixture, (b) The ground truth target source, (c) The separated source queried by AudioCaps description: “**applauding followed by people singing and a tambourine**”, and (d) The separated source queried by our collected human description: “**a show start with audience applauding and then singing**”.

## 6. Conclusions

We have presented a study of the LASS task, which, to our knowledge, is the first attempt bridging audio source separation with natural language queries. We presented a LASS-Net which separates target sources from mixtures using natural language queries. Experimental results show the promising separation results and generalization capabilities of LASS-Net. In future work, we will extend LASS to the scenario of complex acoustic mixtures and fine-grained natural language queries.

## 7. Acknowledgements

This work is supported by a Newton Institutional Links Award from the British Council (Grant number 623805725), an Engineering and Physical Sciences Research Council Grant EP/T019751/1, and a Research Scholarship from the China Scholarship Council. For the purpose of open access, the authors have applied a Creative Commons Attribution (CC BY) licence to any Author Accepted Manuscript version arising.

## 8. References

- [1] D. Yang, H. Wang, Y. Zou, and C. Weng, “Detect what you want: Target sound detection,” *arXiv:2112.10153*, 2021.
- [2] D. Wang and G. J. Brown, *Computational Auditory Scene Analysis: Principles, Algorithms, and Applications*, 2006.
- [3] D. Wang and J. Chen, “Supervised speech separation based on deep learning: An overview,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 26, no. 10, pp. 1702–1726, 2018.
- [4] 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.
- [5] Q. Kong, Y. Cao, H. Liu, K. Choi, and Y. Wang, “Decoupling magnitude and phase estimation with deep resunet for music source separation,” *arXiv:2109.05418*, 2021.
- [6] H. Liu, L. Xie, J. Wu, and G. Yang, “Channel-wise subband input for better voice and accompaniment separation on high resolution music,” in *INTERSPEECH*, 2020.
- [7] H. Liu, Q. Kong, and J. Liu, “CWS-PResUNet: Music source separation with channel-wise subband phase-aware resunet,” *arXiv:2112.04685*, 2021.
- [8] Q. Kong, Y. Wang, X. Song, Y. Cao, W. Wang, and M. D. Plumbley, “Source separation with weakly labelled data: An approach to computational auditory scene analysis,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 101–105.
- [9] I. Kavalero, S. Wisdom, H. Erdogan, B. Patton, K. Wilson, J. Le Roux, and J. R. Hershey, “Universal sound separation,” in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2019, pp. 175–179.
- [10] K. Chen, X. Du, B. Zhu, Z. Ma, T. Berg-Kirkpatrick, and S. Dubnov, “Zero-shot audio source separation through query-based learning from weakly-labeled data,” in *AAAI*, 2022.
- [11] S. Wisdom, H. Erdogan, D. P. Ellis, R. Serizel, N. Turpault, E. Fonseca, J. Salamon, P. Seetharaman, and J. R. Hershey, “What’s all the fuss about free universal sound separation data?” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2021, pp. 186–190.
- [12] E. Tzinis, S. Wisdom, J. R. Hershey, A. Jansen, and D. P. Ellis, “Improving universal sound separation using sound classification,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 96–100.
- [13] T. Ochiai, M. Delcroix, Y. Koizumi, H. Ito, K. Kinoshita, and S. Araki, “Listen to what you want: Neural network-based universal sound selector,” in *INTERSPEECH*, 2020.
- [14] S. Rubin, F. Berthouzoz, G. J. Mysore, W. Li, and M. Agrawala, “Content-based tools for editing audio stories,” in *Proceedings of the 26th annual ACM Symposium on User Interface Software and Technology*, 2013, pp. 113–122.
- [15] A.-M. Oncescu, A. Koepke, J. F. Henriques, Z. Akata, and S. Albanie, “Audio retrieval with natural language queries,” in *INTERSPEECH*, 2021.
- [16] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional Transformers for language understanding,” *arXiv:1810.04805*, 2018.
- [17] 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.
- [18] Q. Wang, H. Muckenhirn, K. Wilson, P. Sridhar, Z. Wu, J. Hershey, R. A. Saurous, R. J. Weiss, Y. Jia, and I. L. Moreno, “VoiceFilter: Targeted voice separation by speaker-conditioned spectrogram masking,” *arXiv:1810.04826*, 2018.
- [19] 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,” in *INTERSPEECH*, 2020.
- [20] Y. Okamoto, S. Horiguchi, M. Yamamoto, K. Imoto, and Y. Kawaguchi, “Environmental sound extraction using onomatopoeia,” *arXiv:2112.00209*, 2021.
- [21] K. Drossos, S. Adavanne, and T. Virtanen, “Automated audio captioning with recurrent neural networks,” in *2017 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, pp. 374–378.
- [22] X. Mei, Q. Huang, X. Liu, G. Chen, J. Wu, Y. Wu, J. Zhao, S. Li, T. Ko, H. L. Tang, M. D. Plumbley, and W. Wang, “An encoder-decoder based audio captioning system with transfer and reinforcement learning,” in *Proceedings of the Detection and Classification of Acoustic Scenes and Events Workshop*, 2021.
- [23] X. Liu, Q. Huang, X. Mei, T. Ko, H. L. Tang, M. D. Plumbley, and W. Wang, “CL4AC: A contrastive loss for audio captioning,” in *Proceedings of the Detection and Classification of Acoustic Scenes and Events Workshop*, 2021.
- [24] X. Mei, X. Liu, J. Sun, M. D. Plumbley, and W. Wang, “Diverse audio captioning via adversarial training,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022.
- [25] K. Drossos, S. Lipping, and T. Virtanen, “Clotho: An audio captioning dataset,” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2020, pp. 736–740.
- [26] X. Mei, X. Liu, Q. Huang, M. D. Plumbley, and W. Wang, “Audio captioning Transformer,” in *Proceedings of the Detection and Classification of Acoustic Scenes and Events Workshop*, 2021.
- [27] X. Liu, X. Mei, Q. Huang, J. Sun, J. Zhao, H. Liu, M. D. Plumbley, V. Kiliç, and W. Wang, “Leveraging pre-trained BERT for audio captioning,” *arXiv:2203.02838*, 2022.
- [28] Y. Zhu, R. Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, and S. Fidler, “Aligning books and movies: Towards story-like visual explanations by watching movies and reading books,” in *Proceedings of the IEEE International Conference on Computer Vision*, 2015, pp. 19–27.
- [29] B. Weck, X. Favory, K. Drossos, and X. Serra, “Evaluating off-the-shelf machine listening and natural language models for automated audio captioning,” *arXiv:2110.07410*, 2021.
- [30] I. Turc, M.-W. Chang, K. Lee, and K. Toutanova, “Well-read students learn better: On the importance of pre-training compact models,” *arXiv:1908.08962*, 2019.
- [31] E. Perez, F. Strub, H. De Vries, V. Dumoulin, and A. Courville, “FiLM: Visual reasoning with a general conditioning layer,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 32, no. 1, 2018.
- [32] 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 *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017, pp. 776–780.
- [33] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, M. Norouzi, W. Macherey, M. Krikun, Y. Cao, Q. Gao, K. Macherey *et al.*, “Google’s neural machine translation system: Bridging the gap between human and machine translation,” *arXiv:1609.08144*, 2016.
- [34] D. P. Kingma and J. Ba, “Adam: A method for stochastic optimization,” *International Conference on Learning Representations (ICLR)*, 2015.
- [35] E. Vincent, R. Gribonval, and C. Févotte, “Performance measurement in blind audio source separation,” *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 14, no. 4, pp. 1462–1469, 2006.
- [36] J. Le Roux, S. Wisdom, H. Erdogan, and J. R. Hershey, “SDR—half-baked or well done?” in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 626–630.