



Leveraging Unlabeled Audio for Audio-Text Contrastive Learning via Audio-Composed Text Features

Tatsuya Komatsu¹, Hokuto Munakata¹, Yuchi Ishikawa¹

¹LY Corporation, Tokyo, Japan

komatsu.tatsuya@lycorp.co.jp, hokuto.munakata@lycorp.co.jp, yuchi.ishikawa@lycorp.co.jp

Abstract

We propose a novel approach to audio-text contrastive learning that leverages unlabeled audio by introducing audio-composed text features. First, we generate composed audio by additively combining labeled and unlabeled audio. To obtain a text feature aligned with this newly composed audio, we introduce an audio-to-text (a2t) module that transforms the features of unlabeled audio into the corresponding text feature. The newly generated text feature is then concatenated with the original text of the labeled audio and passed through a text encoder to produce the audio-composed text features. By pairing these features with the composed audio for contrastive learning, our approach effectively integrates information from both labeled and unlabeled data. In audio-text retrieval experiments on Clotho and AudioCaps, the proposed method achieves notable improvements in Recall@1, with relative gains of 9.3% and 13.6%, respectively, compared to those trained solely with labeled audio. **Index Terms:** contrastive learning, unlabeled audio, audio-composed text features, audio-text retrieval,

1. Introduction

Audio-text contrastive learning, which maps audio and text into the same representation space, has gathered significant attention as a foundational technology supporting a wide range of applications. These include language-based audio retrieval [1, 2, 3], audio captioning [4, 5, 6, 7], text-to-audio generation [8, 9, 10], and audio large language models (LLMs) [11, 12]. In the field of Computer Vision, CLIP [13] (Contrastive Language-Image Pre-training) has already achieved success. Following this model, numerous frameworks such as CLAP [14], WAV2CLIP [15], AudioCLIP [16], and MuLan [17] have been proposed in the audio domain.

However, building joint representations for audio and text typically requires a substantial amount of paired (audio and text) data. For example, CLIP is trained on billion-scale image-text datasets like LAION-5B [18] are used. In contrast, even the most widely used audio-text datasets, Clotho [19] and AudioCaps [20], combined yield only about 50,000 pairs. Consequently, this scarcity of data emerges as a significant bottleneck for model performance in audio-text tasks.

Efforts to address data scarcity often focus on extending existing audio-text pairs or synthesizing descriptions for audio clips using captioning models and large language models (LLMs). Some methods mix two audio clips and concatenate their respective captions using simple conjunctions like “and” [21, 22, 23]. Other methods including T-CLAP [24] employ temporal connectors like “followed by” or “after” to handle temporal dependencies in audio contents [25, 26]. Additionally, datasets such as ClothoV2-GPT [27], WavCaps [28], and Auto-

ACD [29] generate text captions using metadata and audio tags, leveraging LLMs to produce large-scale audio-text data.

While LLM-based caption generation for audio clips holds promise for producing vast amounts of training data, these methods often rely on some form of textual information, such as tags or metadata. Moreover, the use of AI-generated text raises concerns about model collapse [30, 31], model bias [32], and may obscure nuanced audio features or long-tailed information [30, 31, 32]. The generated text may also deviate from human-annotated texts, leading to limited performance gains. Indeed, some studies have shown that simply relying on generated text does not yield substantial improvements [2, 3], highlighting the drawbacks of straightforward text synthesis for data augmentation.

On the other hand, vast amounts of unlabeled audio data can be obtained from the internet and various media sources. Effectively leveraging such unlabeled audio could fundamentally alleviate data scarcity. In the computer vision domain, a notable example of effectively leveraging unlabeled data is Pic2Word [33], which addresses zero-shot composed image retrieval. For instance, it can retrieve an image of a “blue chair” by combining a chair image query with the text “blue.” Pic2Word introduces a mapping network that translates features from unlabeled images into the text space—effectively encoding them as text embeddings—while the main CLIP backbone remains frozen. This concept of mapping unlabeled data into the text domain and encoding it as a text feature could similarly be applied to enhance audio-text contrastive learning.

In this study, we propose a method inspired by Pic2Word to effectively leverage unlabeled audio in audio-text tasks in addition to labeled audio-text pairs. Specifically, we first convert unlabeled audio into audio features, which are then mapped to text features by an audio-to-text (a2t) network. We combine these pseudo-generated text features with the original labeled text features and feed them into a text encoder to produce audio-composed text features. On the audio side, labeled audio is merged with unlabeled audio and transformed into audio features within a contrastive learning framework. Through this process, our method learns a rich joint feature space for audio and text while capitalizing on the diverse acoustic information of unlabeled data. In our experiments on audio-text retrieval, we demonstrate the proposed method effectively leverages unlabeled audio, leading to significant performance improvements compared to models trained solely on labeled audio.

2. Audio-Text Contrastive Learning

We first describe a conventional audio-text contrastive learning method, illustrated in Figure 1-(a). This approach requires labeled audio and text pairs for training. First, the input audio

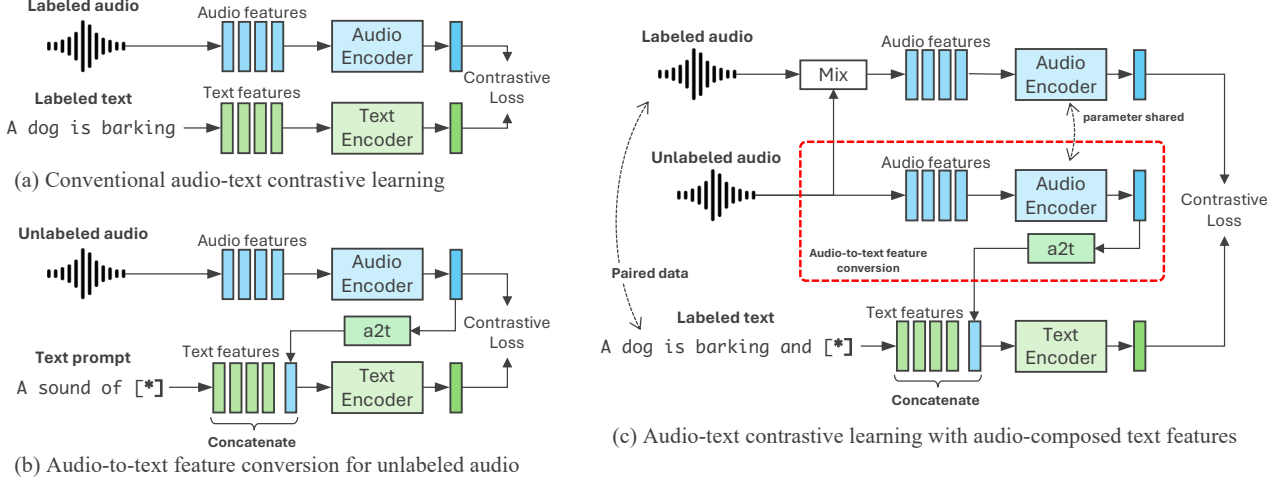


Figure 1: Overview of the conventional audio-text contrastive learning and the proposed contrastive learning with composed text features.

is transformed into frequency-domain features, while the corresponding text is tokenized and converted into text features. Let the audio features be $\mathbf{X}^{\text{audio}} \in \mathbb{R}^{T \times D^{\text{audio}}}$ and the text features be $\mathbf{X}^{\text{text}} \in \mathbb{R}^{L \times D^{\text{text}}}$, where T denotes the number of time frames, D^{audio} is the dimensionality of the audio features, L is the text length, and D^{text} is the dimensionality of the text features. These features are then fed into the respective modality encoders to obtain the audio feature $\mathbf{z}^{\text{audio}}$ and the text feature \mathbf{z}^{text} , both of which share the same dimensionality:

$$\mathbf{z}^{\text{audio}} = \text{AudioEncoder}(\mathbf{X}^{\text{audio}}), \quad (1)$$

$$\mathbf{z}^{\text{text}} = \text{TextEncoder}(\mathbf{X}^{\text{text}}). \quad (2)$$

To optimize the parameters of both encoders, the InfoNCE [13] loss is commonly employed. InfoNCE distinguishes positive (paired) examples from negative (unpaired) examples within a mini-batch. Minimizing InfoNCE effectively pulls matching features closer in the latent space while pushing non-matching pairs farther apart. The InfoNCE loss for the audio-to-text direction is given by:

$$\mathcal{L}_{\text{InfoNCE}}^{\text{a} \rightarrow \text{t}} = -\log \frac{\exp(\cos(\mathbf{z}^{\text{audio}}, \mathbf{z}^{\text{text}})/\tau)}{\sum_{\mathbf{z}'^{\text{text}}} \exp(\cos(\mathbf{z}^{\text{audio}}, \mathbf{z}'^{\text{text}})/\tau)}, \quad (3)$$

where \cos denotes the cosine similarity and $\tau > 0$ is a temperature scaling parameter. Similarly, the InfoNCE loss for the text-to-audio direction is defined as:

$$\mathcal{L}_{\text{InfoNCE}}^{\text{t} \rightarrow \text{a}} = -\log \frac{\exp(\cos(\mathbf{z}^{\text{text}}, \mathbf{z}^{\text{audio}})/\tau)}{\sum_{\mathbf{z}'^{\text{audio}}} \exp(\cos(\mathbf{z}^{\text{text}}, \mathbf{z}'^{\text{audio}})/\tau)}. \quad (4)$$

The total InfoNCE loss is then the sum of both directional losses:

$$\mathcal{L}_{\text{InfoNCE}}(\mathbf{z}^{\text{audio}}, \mathbf{z}^{\text{text}}) = \mathcal{L}_{\text{InfoNCE}}^{\text{t} \rightarrow \text{a}} + \mathcal{L}_{\text{InfoNCE}}^{\text{a} \rightarrow \text{t}}. \quad (5)$$

By minimizing $\mathcal{L}_{\text{InfoNCE}}$, the model learns features that place paired audio and text closer in the latent space, facilitating downstream tasks such as retrieval and zero-shot classification.

3. Proposed Method

In this section, we detail the proposed contrastive learning method that leverages audio-composed text features. By introducing an audio-to-text (a2t) module, our approach enables contrastive learning with unlabeled audio. Specifically, this a2t module projects unlabeled audio embeddings into the text feature space, generating pseudo-paired text embeddings, i.e., audio-composed text features, for contrastive learning.

3.1. Audio-to-Text Feature Conversion for Unlabeled Audio

Figure 1-(b) illustrates how unlabeled audio can be transformed into a text-like feature via the audio-to-text (a2t) module. This module is inspired by Pic2Word [33], which maps image features into text features to compose them with text queries, enabling zero-shot composed image retrieval.

Given an unlabeled audio input $\mathbf{X}_{(\text{unlabel})}^{\text{audio}}$, we first obtain its audio embedding by using the same audio encoder as in the labeled case in Eq.(1):

$$\mathbf{z}_{(\text{unlabel})}^{\text{audio}} = \text{AudioEncoder}(\mathbf{X}_{(\text{unlabel})}^{\text{audio}}). \quad (6)$$

Next, we apply the a2t module, denoted f_{a2t} , to map this audio feature into a text feature space:

$$\mathbf{z}_{(\text{a2t})}^{\text{text}} = f_{\text{a2t}}(\mathbf{z}_{(\text{unlabel})}^{\text{audio}}). \quad (7)$$

We then concatenate the resulting feature with a text prompt $\mathbf{X}_{(\text{prompt})}^{\text{text}}$ along the token axis to construct a pseudo-text input:

$$\hat{\mathbf{X}}^{\text{text}} = [\mathbf{X}_{(\text{prompt})}^{\text{text}}; \mathbf{z}_{(\text{a2t})}^{\text{text}}], \quad (8)$$

where $[\mathbf{A}; \mathbf{B}]$ denotes concatenation of sequences \mathbf{A} and \mathbf{B} along the token axis. Finally, this concatenated representation is fed into the text encoder to obtain the text-domain embedding for unlabeled audio:

$$\hat{\mathbf{z}}^{\text{text}} = \text{TextEncoder}(\hat{\mathbf{X}}^{\text{text}}). \quad (9)$$

By converting unlabeled audio into a text-like representation, we can leverage the same text encoder and subsequent contrastive learning framework used for labeled data. The a2t

module’s parameters are learned with an InfoNCE loss based on Eq. (5):

$$\mathcal{L}_{a2t} = \mathcal{L}_{\text{InfoNCE}}(\mathbf{z}_{(\text{unlabel})}^{\text{audio}}, \hat{\mathbf{z}}^{\text{text}}) \quad (10)$$

Thus, unlabeled audio is effectively brought into a shared feature space, enabling training even when paired labels are unavailable.

3.2. Learning with Audio-Composed Text-Features

By using the a2t module trained in Sec 3.1, we can incorporate unlabeled audio into a contrastive learning framework. In this study, we perform data augmentation by mixing unlabeled audio with labeled audio and deriving corresponding text feature via the a2t module.

Figure 1(c) illustrates the process of composing labeled audio with unlabeled audio to obtain the necessary text features. Let $\mathbf{X}^{\text{audio}}$ be a labeled audio input and $\mathbf{X}_{(\text{unlabel})}^{\text{audio}}$ be an unlabeled audio input, with their respective audio features $\mathbf{z}^{\text{audio}}$ and $\mathbf{z}_{(\text{unlabel})}^{\text{audio}}$ as previously defined. To augment the labeled audio, we mix it with the unlabeled audio:

$$\mathbf{X}_{(\text{composed})}^{\text{audio}} = (\mathbf{X}^{\text{audio}} + \mathbf{X}_{(\text{unlabel})}^{\text{audio}})/2. \quad (11)$$

We then feed this mixture into the audio encoder to obtain the composed audio feature:

$$\mathbf{z}_{(\text{composed})}^{\text{audio}} = \text{AudioEncoder}(\mathbf{X}_{(\text{composed})}^{\text{audio}}). \quad (12)$$

To perform contrastive learning with $\mathbf{z}_{(\text{composed})}^{\text{audio}}$, we need the corresponding text feature. However, part of $\mathbf{z}_{(\text{composed})}^{\text{audio}}$ contains unlabeled audio components. In the proposed method, we estimate these unlabeled components via the a2t module, which maps them into the text feature domain. Specifically, we first encode and transform only the unlabeled audio $\mathbf{X}_{(\text{unlabel})}^{\text{audio}}$ according to Eqs. (6) and (7):

$$\mathbf{z}_{(\text{unlabel})}^{\text{audio}} = \text{AudioEncoder}(\mathbf{X}_{(\text{unlabel})}^{\text{audio}}), \quad (13)$$

$$\mathbf{z}_{(\text{a2t})}^{\text{text}} = f_{\text{a2t}}(\mathbf{z}_{(\text{unlabel})}^{\text{audio}}). \quad (14)$$

Since the a2t module is trained to align $\mathbf{z}_{(\text{a2t})}^{\text{text}}$ with $\mathbf{z}_{(\text{unlabel})}^{\text{audio}}$, as described in Sec. 3.1, $\mathbf{z}_{(\text{a2t})}^{\text{text}}$ can be handled as a text feature containing information of unlabeled audio $\mathbf{X}_{(\text{unlabel})}^{\text{audio}}$. We then concatenate $\mathbf{z}_{(\text{a2t})}^{\text{text}}$ with the labeled text input \mathbf{X}^{text} to construct the text feature for the composed audio $\mathbf{X}_{(\text{composed})}^{\text{audio}}$:

$$\mathbf{X}_{(\text{composed})}^{\text{text}} = [\mathbf{X}^{\text{text}}; \hat{\mathbf{z}}_{(\text{a2t})}^{\text{text}}], \quad (15)$$

Feeding $\mathbf{X}_{(\text{composed})}^{\text{text}}$ into the text encoder yields:

$$\mathbf{z}_{(\text{composed})}^{\text{text}} = \text{TextEncoder}(\mathbf{X}_{(\text{composed})}^{\text{text}}). \quad (16)$$

Finally, we obtain the composed audio and text pair $(\mathbf{z}_{(\text{composed})}^{\text{audio}}, \mathbf{z}_{(\text{composed})}^{\text{text}})$, allowing us to compute the InfoNCE loss in the same manner as Eqs. (5) and (10):

$$\mathcal{L}_{\text{composed}} = \text{InfoNCE}(\mathbf{z}_{(\text{composed})}^{\text{audio}}, \mathbf{z}_{(\text{composed})}^{\text{text}}). \quad (17)$$

The overall loss function of the proposed method is combination of Eqs. (5), (10), and (17)

$$\mathcal{L}_{\text{proposed}} = \mathcal{L}_{\text{InfoNCE}} + \mathcal{L}_{\text{a2t}} + \mathcal{L}_{\text{composed}}. \quad (18)$$

The first term trains the joint encoders using labeled text–audio pairs, the second term trains the a2t module on unlabeled audio,

and the third term refines the joint encoders by incorporating the composition of labeled and unlabeled audio.

Through this process, unlabeled audio is transformed into text features and integrated with existing text features, enabling contrastive learning even in the absence of paired labels.

4. Experiments

To validate the effectiveness of the proposed method, we conduct experiments on audio-text retrieval tasks. In these experiments, we assume a scenario in which only the target dataset provides labeled audio-text pairs, while large amounts of unlabeled audio from other datasets are available. We specifically investigate whether our audio-composed text features can exploit these unlabeled audio clips to improve retrieval performance in this limited-label setting.

4.1. Datasets

We use two widely adopted audio-text datasets, Clotho [19] and AudioCaps [20], as our main sources of labeled data. Additionally, the balanced train split of AudioSet [34] is employed as an unlabeled dataset. In each experiment, we designate either Clotho or AudioCaps as the retrieval target, while treating the remaining dataset (as well as AudioSet) purely as a source of unlabeled audio clips. For instance, when Clotho is the target, AudioCaps and AudioSet are used as unlabeled audio. Conversely, when AudioCaps is the target, Clotho and AudioSet are used as unlabeled audio.

We measure retrieval performance using Recall@1, Recall@5, Recall@10, and mAP@10 for both Audio-to-Text and Text-to-Audio retrieval. These metrics represent how effectively the model aligns audio and text embeddings under each experimental setting.

4.2. Network Architecture and Training

We adopt CLAP (ms-clap;2023) [35] as our base audio-text model for both the baseline model and proposed method. The a2t module in the proposed method is inspired by the architecture from Pic2Word [33], comprising a two-layer MLP (Multi Layer Perceptron) including a fully connected layer, a dropout layer, and a ReLU activation function. The fully connected layers transform the data while maintaining a consistent dimension of 1024, matching the output dimension of CLAP encoders. Following the MLPs, there is an output layer with a single fully connected layer, which projects the output to the text-token feature.

The proposed method is trained using the loss function defined in Eq. (18). The first term focuses on labeled audio to ensure effective learning from audio samples with associated labels. The second term addresses unlabeled audio, using the text prompt “this sound is” for computing text features, as in Eq. (8). The third term is computed using mixtures of labeled and unlabeled audio samples with their corresponding composed texts, aiming to leverage both data types for robust training.

The number of epochs for fine-tuning is 30, using the parameters from the final epoch for evaluation. The batch size is set to 256, and training is conducted on four NVIDIA A100 GPUs, requiring approximately one to two hours per experiment. We employ the Adam optimizer [36] along with a cosine-annealing scheduler for the learning rate. The settings are consistent across all experiments.

Table 1: *Retrieval results on the Clotho and AudioCaps datasets. Recall@1, Recall@5, Recall@10, and mAP@10 for both text → audio and audio → text tasks are reported. “A+T” indicates the usage of audio-text pairs for training, “A” indicates the use of only audio as unlabeled data, and “-” indicates that the dataset was not used.*

Retrieval performance of the Clotho dataset											
Method	Training data			text → audio retrieval				audio → text retrieval			
	Clotho	AudioCaps	AudioSet	R@1	R@5	R@10	mAP@10	R@1	R@5	R@10	mAP@10
vanilla CLAP	-	-	-	15.75	39.94	52.27	25.99	17.22	40.86	55.69	27.71
finetune	A+T	-	-	16.98	40.69	54.12	27.27	15.89	39.9	52.63	26.11
<i>Proposed method</i>											
composed	A+T	Audio	-	18.56	42.78	57.03	29.15	19.43	42.58	56.75	29.07
composed	A+T	-	Audio	18.41	43.73	57.67	29.21	17.99	41.91	56.08	28.46
<i>(Scenario where all paired data is available)</i>											
finetune [†]	A+T	A+T	-	17.28	41.01	54.64	27.66	15.98	38.56	53.49	26.15
concatenate [‡]	A+T	A+T	-	18.01	44.34	57.22	29.16	19.9	42.68	56.84	29.8
Retrieval performance of the AudioCaps dataset											
Method	Training data			text → audio retrieval				audio → text retrieval			
	Clotho	AudioCaps	AudioSet	R@1	R@5	R@10	mAP@10	R@1	R@5	R@10	mAP@10
vanilla CLAP	-	-	-	15.79	46.21	62.94	28.67	25.25	57.3	74.75	39.19
finetune	-	A+T	-	33.5	68.2	81.11	48.11	33.98	68.97	81.31	48.5
<i>Proposed method</i>											
composed	Audio	A+T	-	38.05	73.57	85.87	53.12	40.54	76.56	86.75	55.27
composed	-	A+T	Audio	37.3	72.96	85.44	52.58	40.32	74.41	83.92	54.6
<i>(Scenario where all paired data is available)</i>											
finetune [†]	A+T	A+T	-	35.08	70.53	83.28	50.11	36.24	71.23	84.03	50.81
concatenate [‡]	A+T	A+T	-	38.26	72.55	85.03	52.91	40.32	73.27	85.16	54.21

[†] Fine-tuned using paired audio-text data from both Clotho and AudioCaps.

[‡] Mixup-like data augmentation: composing two audio as the proposed method while concatenating their corresponding texts.

4.3. Experimental results

Table 1 summarizes the results of audio-text retrieval (text→audio and audio→text) when Clotho or AudioCaps is used as the target dataset. Despite leveraging only *unlabeled* audio from the other datasets, our proposed method with *audio-composed text* (denoted as “composed” in the table) consistently outperforms the baselines.

First, in the top part of the table (where Clotho is the target), we observe that our proposed “composed” significantly improves metrics such as Recall@1, Recall@5, and mAP@10 compared to “vanilla CLAP,” which directly uses CLAP without fine-tuning, and “finetune,” which is fine-tuned on Clotho-labeled pairs only.

Similarly, in the bottom part of the table (where AudioCaps is the target), the proposed methods “composed” also show marked improvements over “vanilla CLAP” and fine-tuning on AudioCaps alone.

Furthermore, the table includes results under an ideal scenario where labeled pairs from both Clotho and AudioCaps are available (e.g., “finetune[†]” and “concatenate[‡]”). As expected, having more labeled data generally boosts performance, and in some cases, a simple concatenation (e.g., “concatenate[‡]”) yields high accuracy. Nonetheless, it is particularly noteworthy that *our proposed method still outperforms these approaches, even when all labeled data from both datasets are leveraged*. This underscores the robustness of our method, which effectively exploits unlabeled audio to achieve superior performance, even when no additional labeled pairs can be obtained from non-target datasets.

These findings demonstrate that the proposed audio-composed text approach significantly improves retrieval metrics

even when no labels are available for additional audio outside the target dataset.

5. Conclusion

In this paper, we introduced a novel audio-text contrastive learning framework that effectively leverages unlabeled audio through the concept of audio-composed text features. Rather than relying on manually annotated or synthesized captions for the unlabeled data, we proposed an audio-to-text (a2t) module that converts unlabeled audio features into text-like features. By concatenating these features with text features from labeled data, we can form composed text features and use them alongside composed audio for contrastive training. This strategy enriches the model’s feature space by integrating diverse information from both labeled and unlabeled audio signals.

Experimental evaluations on Clotho and AudioCaps demonstrated that our method achieves marked improvements in audio-text retrieval performance, notably boosting Recall@1 by up to 9.3% and 13.6% relative to baselines. Crucially, these gains were realized without additional text annotations for the unlabeled audio, thus underscoring the approach’s potential in scenarios where large amounts of unlabeled audio data are readily available but paired captions are scarce. Moreover, it is noteworthy that our proposed method outperforms conventional approaches even in an ideal scenario where labeled pairs from both Clotho and AudioCaps are available.

6. References

- [1] H. Xie, S. Lipping, and T. Virtanen, “Language-based audio retrieval task in dcase 2022 challenge,” *arXiv preprint arXiv:2206.06108*, 2022.

- [2] H. Munakata, T. Nishimura, S. Nakada, and T. Komatsu, "Training strategy of massive text-to-audio models and gpt-based query-augmentation," DCASE2024 Challenge, Tech. Rep., June 2024.
- [3] P. Primus and G. Widmer, "A knowledge distillation approach to improving language-based audio retrieval models," DCASE2024 Challenge, Tech. Rep., June 2024.
- [4] K. Drossos, S. Adavanne, and T. Virtanen, "Automated audio captioning with recurrent neural networks," in *Proc. WASPAA2017*, 2017, pp. 374–378.
- [5] M. Wu, H. Dinkel, and K. Yu, "Audio caption: Listen and tell," in *Proc. ICASSP2019*, 2019, pp. 830–834.
- [6] X. Mei, X. Liu, Q. Huang, M. D. Plumbley, and W. Wang, "Audio captioning transformer," in *Proc. DCASE2021*, Barcelona, Spain, November 2021, pp. 211–215.
- [7] X. Mei, X. Liu, M. D. Plumbley, and W. Wang, "Automated audio captioning: An overview of recent progress and new challenges," *EURASIP J. Audio Speech Music Process.*, vol. 2022, no. 1, oct 2022. [Online]. Available: <https://doi.org/10.1186/s13636-022-00259-2>
- [8] H. Liu, Z. Chen, Y. Yuan, X. Mei, X. Liu, D. Mandic, W. Wang, and M. D. Plumbley, "Audioldm: Text-to-audio generation with latent diffusion models," in *Proc. ICML*. PMLR, 2023, pp. 21 450–21 474.
- [9] D. Ghosal, N. Majumder, A. Mehrish, and S. Poria, "Text-to-audio generation using instruction guided latent diffusion model," in *Proc. ICMM*. New York, NY, USA: Association for Computing Machinery, 2023, p. 3590–3598.
- [10] F. Kreuk, G. Synnaeve, A. Polyak, U. Singer, A. Défossez, J. Copet, D. Parikh, Y. Taigman, and Y. Adi, "Audiogen: Textually guided audio generation," in *Proc. ICLR*, 2023.
- [11] Y. Chu, J. Xu, X. Zhou, Q. Yang, S. Zhang, Z. Yan, C. Zhou, and J. Zhou, "Qwen-audio: Advancing universal audio understanding via unified large-scale audio-language models," *arXiv preprint arXiv:2311.07919*, 2023.
- [12] S. Ghosh, S. Kumar, A. Seth, C. K. R. Evuru, U. Tyagi, S. Sakshi, O. Nieto, R. Duraiswami, and D. Manocha, "Gama: A large audio-language model with advanced audio understanding and complex reasoning abilities," *arXiv preprint arXiv:2406.11768*, 2024.
- [13] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *Proc. ICML*. PMLR, 2021, pp. 8748–8763.
- [14] B. Elizalde, S. Deshmukh, M. Al Ismail, and H. Wang, "Clap learning audio concepts from natural language supervision," in *Proc. ICASSP*. IEEE, 2023, pp. 1–5.
- [15] H.-H. Wu, P. Seetharaman, K. Kumar, and J. P. Bello, "Wav2clip: Learning robust audio representations from clip," in *ICASSP 2022-2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2022, pp. 4563–4567.
- [16] A. Guzhov, F. Raue, J. Hees, and A. Dengel, "Audioclip: Extending clip to image, text and audio," in *Proc. ICASSP*. IEEE, 2022, pp. 976–980.
- [17] Q. Huang, A. Jansen, J. Lee, R. Ganti, J. Y. Li, and D. P. Ellis, "Mulan: A joint embedding of music audio and natural language," in *Proc. ISMIR*, 2022.
- [18] C. Schuhmann, R. Beaumont, R. Vencu, C. Gordon, R. Wightman, M. Cherti, T. Coombes, A. Katta, C. Mullis, M. Wortsman *et al.*, "Laion-5b: An open large-scale dataset for training next generation image-text models," *Proc. NeurIPS*, vol. 35, pp. 25 278–25 294, 2022.
- [19] K. Drossos, S. Lipping, and T. Virtanen, "Clotho: An audio captioning dataset," in *Proc. ICASSP*, 2020, pp. 736–740.
- [20] C. D. Kim, B. Kim, H. Lee, and G. Kim, "AudioCaps: Generating captions for audios in the wild," in *Proc. NAACL*, 2019.
- [21] E. Kim, J. Kim, Y. Oh, K. Kim, M. Park, J. Sim, J. Lee, and K. Lee, "Exploring train and test-time augmentations for audio-language learning," *arXiv preprint arXiv:2210.17143*, 2023.
- [22] J.-H. Cho, Y.-A. Park, J. Kim, and J.-H. Chang, "Hyu submission for the dcase 2023 task 6a: automated audio captioning model using al-mixgen and synonyms substitution," DCASE2023 Challenge, Tech. Rep., May 2023.
- [23] J. Kim, Y.-A. Park, J.-H. Cho, and J.-H. Chang, "Improving automated audio captioning fluency through data augmentation and ensemble selection," in *Proc. DCASE2023*, Tampere, Finland, September 2023, pp. 86–90.
- [24] Y. Yuan, Z. Chen, X. Liu, H. Liu, X. Xu, D. Jia, Y. Chen, M. D. Plumbley, and W. Wang, "T-clap: Temporal-enhanced contrastive language-audio pretraining," *arXiv preprint arXiv:2404.17806*, 2024.
- [25] H.-H. Wu, O. Nieto, J. P. Bello, and J. Salamon, "Audio-text models do not yet leverage natural language," in *Proc. ICASSP 2023*, 2023, pp. 1–5.
- [26] Z. Xie, X. Xu, M. Wu, and K. Yu, "Enhance Temporal Relations in Audio Captioning with Sound Event Detection," in *Proc. INTERSPEECH 2023*, 2023, pp. 4179–4183.
- [27] P. Primus, K. Koutini, and G. Widmer, "Advancing natural-language based audio retrieval with passt and large audio-caption data sets," in *Proc. DCASE 2023 Workshop*, 2023.
- [28] 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.
- [29] L. Sun, X. Xu, M. Wu, and W. Xie, "A large-scale dataset for audio-language representation learning," *arXiv preprint arXiv:2309.11500*, 2023.
- [30] I. Shumailov, Z. Shumaylov, Y. Zhao, N. Papernot, R. Anderson, and Y. Gal, "Ai models collapse when trained on recursively generated data," *Nature*, vol. 631, no. 8022, pp. 755–759, 2024.
- [31] I. Shumailov, Z. Shumaylov, Y. Zhao, Y. Gal, N. Papernot, and R. Anderson, "The curse of recursion: Training on generated data makes models forget," *arXiv preprint arXiv:2305.17493*, 2023.
- [32] S. Wyllie, I. Shumailov, and N. Papernot, "Fairness feedback loops: training on synthetic data amplifies bias," in *The 2024 ACM Conference on Fairness, Accountability, and Transparency*, 2024, pp. 2113–2147.
- [33] K. Saito, K. Sohn, X. Zhang, C.-L. Li, C.-Y. Lee, K. Saenko, and T. Pfister, "Pic2word: Mapping pictures to words for zero-shot composed image retrieval," in *Proc. CVPR*, 2023, pp. 19 305–19 314.
- [34] J. F. Gemmeke, D. P. W. 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 *Proc. ICASSP*, 2017, pp. 776–780.
- [35] B. Elizalde, S. Deshmukh, and H. Wang, "Natural language supervision for general-purpose audio representations," *arXiv preprint arXiv:2309.05767*, 2023.
- [36] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *Proc. ICLR*, 2015.