LARGE-SCALE CONTRASTIVE LANGUAGE-AUDIO PRETRAINING WITH FEATURE FUSION AND KEYWORD-TO-CAPTION AUGMENTATION

Yusong Wu [1]*, *Ke Chen* [2]*, *Tianyu Zhang* [1]*, *Yuchen Hui* [1,3]*, *Taylor Berg-Kirkpatrick* [2], *Shlomo Dubnov* [2]

[1]Mila, Quebec Artificial Intelligence Institute, Université de Montréal
[2]University of California San Diego [3]LAION

ABSTRACT

Contrastive learning has shown remarkable success in the field of multimodal representation learning. In this paper, we propose a pipeline of contrastive language-audio pretraining to develop an audio representation by combining audio data with natural language descriptions. To accomplish this target, we first release LAION-Audio-630K, a large collection of 633,526 audio-text pairs from different data sources. Second, we construct a contrastive language-audio pretraining model by considering different audio encoders and text encoders. We incorporate the feature fusion mechanism and keyword-to-caption augmentation into the model design to further enable the model to process audio inputs of variable lengths and enhance the performance. Third, we perform comprehensive experiments to evaluate our model across three tasks: text-to-audio retrieval, zero-shot audio classification, and supervised audio classification. The results demonstrate that our model achieves superior performance in text-to-audio retrieval task. In audio classification tasks, the model achieves state-of-the-art performance in the zero-shot setting and is able to obtain performance comparable to models' results in the non-zero-shot setting. LAION-Audio-630K[1] and the proposed model[2] are both available to the public.

Index Terms**—** Contrastive Learning, Representation Learning, Text-to-Audio Retrieval, Audio Classification, Audio Dataset

1. INTRODUCTION

Audio is one of the most common information types in the world alongside text and image data. However, different audio tasks typically require finely-annotated data, which limits the amount of available audio data due to the labor-intensive collection procedure. Consequently, designing an effective audio representation for many audio tasks without requiring a lot of supervision remains a challenge.

The contrastive learning paradigm is a successful solution for training a model on large-scale noisy data collected from internet. The recently proposed Contrastive Language-Image Pretraining (CLIP) [1] learns the correspondence between text and image by projecting them into a shared latent space. The training is conducted by regarding the ground-truth image-text pair as the positive sample and left as negative. In contrast to training on unimodal data, CLIP is not constrained by data annotation and shows great robustness by achieving high accuracy in a zero-shot setting on out-of-domain variations of ImageNet dataset [2]. Additionally, CLIP shows great success in downstream tasks such as text-to-image retrieval and text-guided captioning. Similar to vision, audio and natural languages also contain overlapping information. In audio event classification task, for instance, some text descriptions of an event can be mapped to the corresponding audio. These text descriptions share a similar meaning that could be learned together with the related audio to form an audio representation of crossmodal information. Additionally, training such a model requires simply paired audio and text data, which is easy to collect.

Several recent studies [3–9] have presented the prototype of the contrastive language-audio pretraining model for the text-to-audio retrieval task. [6] utilizes Pretrained Audio Neural Network (PANN) [10] as the audio encoder, BERT [11] as the text encoder, and several loss functions to evaluate the text-to-audio retrieval performance. [5] further ensemble HTSAT [12] and RoBERTa [13] into the encoder list to further enhance performance. Then, [4] investigates the effectiveness of the learned representation in the downstream task of audio classification. Some other studies, such as AudioClip [3] and WaveCLIP [9], focus more on the contrastive image-audio (or image-audio-language) pretraining model. All these models show great potential for contrastive learning in the audio domain.

Nonetheless, current studies have not shown the full strength of the language-audio contrastive learning. First, the models mentioned above are trained on relatively small datasets, showing that large-scale data collection and augmentation for training are needed. Second, prior work lacks a full investigation of selections and hyperparameter settings of audio/text encoders, which is essential for determining the basic contrastive language-audio architecture. Third, the model struggles to accommodate varied audio lengths, particularly for the transformer-based audio encoder. There should be a solution to handle audio inputs of variable-length. Finally, the majority of language-audio model studies focuses solely on text-to-audio retrieval without assessing their audio representations in downstream tasks. As a representation model, we expect more discoveries of its generalization ability to more downstream tasks.

In this paper, based on our previous work [6], we make contributions to improve the dataset, model design and the experiment setting from above concerns:

- We release LAION-Audio-630K, currently the largest public audio caption dataset of 633,526 audio-text pairs. To facilitate the learning process, we employ the keyword-to-caption model to augment labels of AudioSet [14] into corresponding captions. This dataset can also contribute to other audio tasks.
- We construct a pipeline of contrastive language-audio pretraining. Two audio encoders and three text encoders are selected for testing. We employ feature fusion mechanisms to enhance the performance and enable our model to handle variable-length inputs.
- We conduct comprehensive experiments on the model, including the text-to-audio retrieval task, as well as zero-shot and supervised

*The first four authors have equal contribution. We would like to thank IRCAM — Project REACH for supporting this project.

[1] Dataset: https://github.com/LAION-AI/audio-dataset/

[2] Model: https://github.com/LAION-AI/CLAP

Dataset	Pairs	Audio Durations (hrs)
Clotho [15]	5,929	37.00
SoundDescs [16]	32,979	1060.40
AudioCaps [17]	52,904	144.94
LAION-Audio-630K	633,526	4325.39

Table 1: LAION-Audio-630K compared with existing audio caption datasets.

audio classification downstream tasks. We demonstrate that scaling of the dataset, keyword-to-caption augmentation, and feature fusion can improve the model's performance in different perspectives. It achieves the state-of-the-art (SOTA) in the text-to-audio retrieval and audio classification tasks, even comparable to the performance of supervised models.

We make both LAION-Audio-630K and the proposed model available to the public.

2. LAION-AUDIO-630K AND TRAINING DATASET

2.1. LAION-Audio-630K

We collect LAION-Audio-630K, a large-scale audio-text dataset consisting of 633,526 pairs with the total duration of 4,325.39 hours. It contains audios of human activities, natural sounds and audio effects, consisting of 8 data sources from publicly available websites[3]. We collect these datasets by downloading audios and relevant text descriptions. Based on our current knowledge, LAION-Audio-630K is the largest audio-text dataset publicly available and a magnitude larger than previous audio-text datasets as shown in Table 1.

2.2. Training Dataset

To test how model performance will scale on different sizes and types of dataset, we use three training set setting in the paper, varying from small to large size. These settings employ three datasets: 1) **AudioCaps+Clotho** (**AC+CL**) [15, 17] contains about 55K training samples of audio-text pairs. 2) LAION-Audio-630K (**LA.**) consists of around 630K audio-text pairs. 3) **Audioset** [14] consists of 1.9 million audio samples with only labels available for each sample. When processing these datasets, we exclude all overlapping data in evaluation sets. More details of the training datasets can be found at the online appendix.

2.3. Dataset Format and Preprocessing

All audio files used in this work are preprocessed to mono channel at a sample rate of 48kHz in FLAC format. For datasets with only tags or labels available, we extend labels into captions using the template "The sound of `label-1`, `label-2`, ..., and `label-n`" or the keyword-to-caption model (detail in section 3.5). As a result, we can leverage more data into the training of the contrastive language-audio pretraining model. Combining all the datasets, we increase the total number of audio samples with text caption to 2.5 million.

3. MODEL ARCHITECTURE

3.1. Contrastive Language-Audio Pretraining

Figure 1 depicts the general architecture of our proposed contrastive language-audio encoder model. Similar to CLIP [1], we have two encoders to separately process the input of audio data X_i^a and text data X_i^t, where (X_i^a, X_i^t) is one of audio-text pairs indexed by i. The audio embedding E_i^a and the text embedding E_i^t are respectively obtained by the audio encoder $f_{audio}(\cdot)$ and the text encoder $f_{text}(\cdot)$, with projection layers:

$$E_i^a = MLP_{audio}(f_{audio}(X_i^a)) \tag{1}$$

$$E_i^t = MLP_{text}(f_{text}(X_i^t)) \tag{2}$$

Where the audio/text projection layer is a 2-layer multilayer perceptron (MLP) with ReLU [18] as the activation function to map the encoder outputs into the same dimension D (i.e., $E_i^a, E_i^t \in \mathbb{R}^D$).

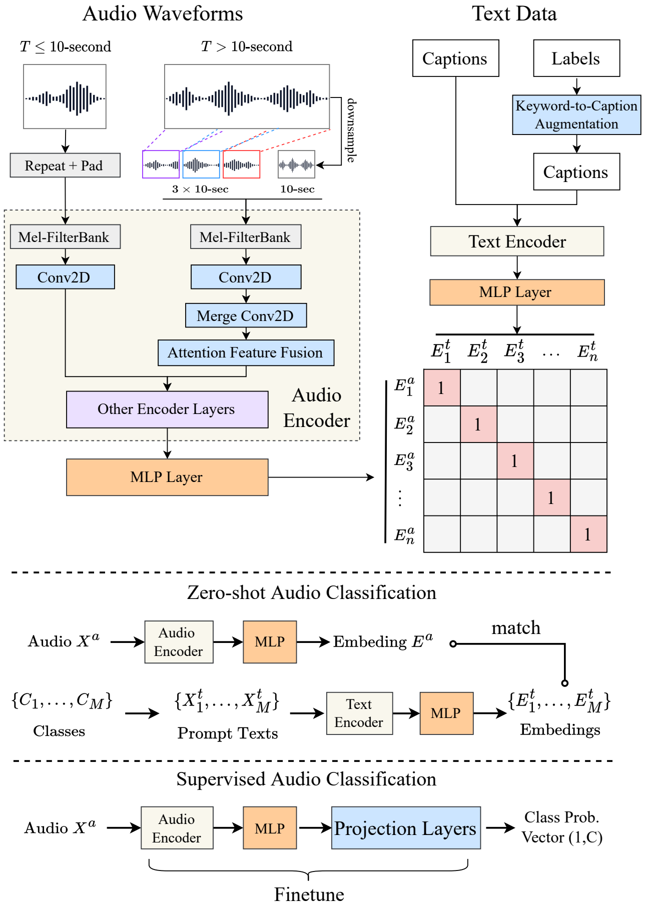


Fig. 1: The architecture of our proposed model, including audio/text encoders, feature fusion, and keyword-to-caption augmentation.

The model is trained with the contrastive learning paradigm between the audio and text embeddings in pair, following the same loss function in [1]:

$$L = \frac{1}{2N} \sum_{i=1}^{N} \left(\log \frac{\exp(E_i^a \cdot E_i^t / \tau)}{\sum_{j=1}^{N} \exp(E_i^a \cdot E_j^t / \tau)} + \log \frac{\exp(E_i^t \cdot E_i^a / \tau)}{\sum_{j=1}^{N} \exp(E_i^t \cdot E_j^a / \tau)} \right) \tag{3}$$

Where τ is a learnable temperature parameter for scaling the loss. Two logarithmic terms consider either audio-to-text logits or text-to-audio logits. N is usually the number of data, but during the training phase, N is used as the batch size, as we cannot compute the whole matrix of all data but update the model by batch gradient descent.

After we train the model, the embeddings (E^a, E^b) can be used for different tasks as shown in Figure 1 and listed in the below subsection.

[3] Dataset details are appended at: https://retrocirce.github.io/appendix/

3.2. Downstream Tasks in Inference Stage

Text-to-Audio Retrieval The target audio embedding E_p^a can find the nearest text embedding E_q^t among M texts $E^t = \{E_1^t, ..., E_M^t\}$ by the cosine similarity function, determining the best match.

Zero-shot Audio Classification For M audio classes $C = \{C_1, ..., C_M\}$, we can construct M prompt texts $X^t = \{X_1^t, ..., X_M^t\}$ (e.g., "the sound of `class-name`"). For a given audio X_p^a, we determine the best match X_q^t among X^t by the cosine similarity function over their embeddings. One advantage of using the contrastive language-audio pretraining is that the categories of audio are unrestricted (i.e., zero-shot) since the model can convert the classification task into the text-to-audio retrieval task.

Supervised Audio Classification After training the model, for a given audio X_p^a, its embedding E_p^a can be further mapped into a fixed-category classification task by adding a projection layer at the back and finetuning (i.e., the non-zero-shot setting).

3.3. Audio Encoders and Text Encoders

We select two models, PANN [10] and HTSAT [12], to construct the audio encoder. PANN is a CNN-based [19] audio classification model with 7 downsampling CNN blocks and 7 upsampling blocks. HTSAT is a transformer-based model with 4 groups of swin-transformer blocks [20], which achieves SOTAs on three audio classification datasets. For both of them, we use their penultimate layer's output, a L-dimension vector as the output sent to the projection MLP layer, where $L_{PANN} = 2048$ and $L_{HTSAT} = 768$.

We select three models, CLIP transformer [1] (text encoder of CLIP), BERT [11], and RoBERTa [13], to construct the text encoder. The output dimension of text encoders is respectively $L_{CLIP} = 512$. $L_{BERT} = 768$, and $L_{RoBERTa} = 768$. We apply both 2-layer MLPs with ReLU activation [18] to map both audio and text outputs into 512 dimensions, which is the size of audio/text representations when training with the contrastive learning paradigm.

3.4. Feature Fusion for Variable-Length Audio

Unlike RGB image data that can be resized to a unified resolution, audio has a nature of variable length. Conventionally, one would input the full audio into the audio encoder and take the average of per-frame or per-chunk audio embeddings as output (i.e., slice & vote). However, the conventional method is computationally inefficient on long audio. As shown in the left of Figure 1, we train and inference on different lengths of audio inputs in constant computation time by combining both coarsely global and randomly sampled local information. For an audio in T seconds and a fixed chunk duration $d = 10$ seconds:

- $T \leq d$: we first repeat the input, then pad it with zero values. For example, a 3-second input will be repeated as $3 \times 3 = 9$-second and padded with 1-second zero values.
- $T > d$: we first downsample the input from T to d-second as a global input. Then we randomly slice three d-second clips, respectively from the front $\frac{1}{3}$, middle $\frac{1}{3}$ and back $\frac{1}{3}$ of the input, as local inputs. We send these $4 \times d$ inputs into the first layer of audio encoder to get the initial features, then three local features will be further converted to one feature by another 2D-Convolution layer with 3-stride in the time axis. Finally, the local feature X_{local}^a and the global feature X_{global}^a will be fused as:

$$X_{fusion}^a = \alpha X_{global}^a + (1 - \alpha) X_{local}^a \tag{4}$$

Where $\alpha = f_{AFF}(X_{global}^a, X_{local}^a)$ is a factor obtained by attention feature fusion (AFF) [21], a two-branch CNN model for learning the fusion factor of two inputs. Comparing with the "slice & vote" method, the feature fusion also saves the training time as we only process audio slices in the first few layers.

Model	AudioCaps (mAP@10)		Clotho (mAP@10)	
	A→T	T→A	A→T	T→A
PANN+CLIP Trans.	4.7	11.7	1.9	4.4
PANN+BERT	34.3	44.3	10.8	17.7
PANN+RoBERTa	37.5	45.3	11.3	18.4
HTSAT+CLIP Trans.	2.4	6.0	1.1	3.2
HTSAT+BERT	43.7	49.2	**13.8**	**20.8**
HTSAT+RoBERTa	**45.7**	**51.3**	**13.8**	20.4

Table 2: The text-to-audio retrieval result (mAP@10) of using different audio/text encoder on AudioCaps and Clotho.

3.5. Keyword-to-Caption Augmentation

As mentioned in section 2.1, some datasets contains reasonable labels or tags as keywords of the corresponding audios. As shown in the right of Figure 1, we used a pre-trained language model T5 [22] to make captions on top of these keywords. We also de-bias the output sentence as post-processing. For example, we replace "woman" and "man" with 'person' as gender de-biasing. Due to the page limit, we provide examples of the augmentation in the online appendix.

4. EXPERIMENTS

In this section, we conduct three experiments on our proposed model. First, we train with different audio and text encoders to find the best baseline combination. Then, we train our model on various dataset size, with the feature fusion and keyword-to-caption augmentation to verify the efficacy of the proposed methods. For the first two experiments, we evaluate our model's performance via recall and mean average precision (mAP) on audio-to-text and text-to-audio retrieval. Lastly, we use the best model to conduct zero-shot and supervised audio classification experiments to evaluate the generalization ability to the downstream tasks.

4.1. Hyperparameters and Training Details

As mentioned in section 2.2, we use AudioCaps, Clotho, LAION-Audio-630K, along with the additional dataset — AudioSet by keyword-to-caption augmentation, to train our model. For the audio data, we use 10-second input length, 480 hop size, 1024 window size, 64 mel-bins to compute STFTs and mel-spectrograms. As the result, each input sent to the audio encoder is of the shape $(T = 1024, F = 64)$. For the text data, we tokenize the text with a maximum token length of 77.

When training the model without the feature fusion, the audio longer than 10-second will be randomly chunked to a 10-second segment. During training, we use the Adam [23] optimizer with $\beta_1 = 0.99$, $\beta_2 = 0.9$ with a warm-up [24] and cosine learning rate decay at a basic learning rate of 10^{-4}. We train the model using a batch size of 768 on **AudioCaps+Clotho** dataset, 2304 on training dataset containing LAION-Audio-630K, and 4608 on training dataset containing **AudioSet**. We train the model for 45 epochs.

4.2. Text-to-Audio Retrieval

Audio and Text Encoders We first conduct experiments to choose the best audio encoder and text encoder for the text-to-audio retrieval task. We combine two audio encoders with three text encoders in section 3.3 where both are loaded from pretrained checkpoints as the same to [5,7,8]. In this experiment, we only train on AudioCaps

Model	Training Set	AudioCaps Eval.						Clotho Eval.					
		T-A Retrieval			A-T Retrieval			T-A Retrieval			A-T Retrieval		
		R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10	R@1	R@5	R@10
MMT [7]	AudioCaps or Clotho	36.1	**72.0**	**84.5**	39.6	76.8	86.7	6.7	21.6	33.2	7.0	22.7	34.6
ML-ACT [8]	AudioCaps or Clotho	33.9	69.7	82.6	39.4	72.0	83.9	14.4	36.6	49.9	16.2	37.6	50.2
CLAP-HTSAT [5]	AudioCaps + Clotho + WT5K	34.6	70.2	82.0	41.9	73.1	84.6	16.7	41.1	54.1	20.0	44.9	58.7
HTSAT-RoBERTa	AudioCaps + Clotho	**36.7**	70.9	83.2	45.3	78.0	87.7	12.0	31.6	43.9	15.7	36.9	51.3
HTSAT-RoBERTa	AudioCaps + Clotho + LA.	32.7	68.0	81.2	43.9	77.7	87.6	15.6	38.6	52.3	23.7	48.9	59.9
HTSAT-RoBERTa (fusion)	AudioCaps + Clotho + LA.	36.2	70.3	82.5	45.0	76.7	88.0	**17.2**	**42.9**	**55.4**	24.2	**51.1**	**66.9**
HTSAT-RoBERTa	ACaps. + Clotho + LA. + AudioSet (template)	34.7	70.5	83.2	45.3	79.5	89.2	16.4	39.0	51.0	21.8	44.6	60.1
HTSAT-RoBERTa	ACaps. + Clotho + LA. + AudioSet (K2C aug.)	36.1	71.8	83.9	**46.8**	**82.9**	**90.7**	16.1	38.3	51.1	22.7	48.5	60.8
HTSAT-RoBERTa (fusion)	ACaps. + Clotho + LA. + AudioSet (K2C aug.)	35.1	71.9	83.7	44.2	80.8	90.3	16.9	41.6	54.4	**24.4**	49.3	65.7

Table 3: The text-to-audio retrieval performance on AudioCaps and Clotho datasets, where "LA." refers to LAION-Audio-630K, "template" refers to the text prompting by templates, "K2C aug." refers to the keyword-to-caption augmentation, and "fusion" refers to the feature fusion.

and Clotho datasets (∼55K data), and report the best mAP@10 on audio-to-text (A→T) and text-to-audio (T→A) perspectives.

According to the results in Table 2, for audio encoder, HTSAT performs better than PANN combined with the RoBERTa or BERT text encoder. For the text encoder, RoBERTa achieves better performance than BERT while the CLIP transformer performs the extremely worst. This coincides with the choice of text encoder in previous works [4, 8]. When further analyzing the loss convergence trends of CLIP transformer model, we find that RoBERTa is less over-fitting, while CLIP transformer is of high-over-fitting, thus resulting its low generalization performance.

Dataset Scale Consequently, we apply HTSAT-RoBERTa as our best model setting to conduct the text-to-audio retrieval experiments as a comprehensive evaluation in Table 3. We adopt the same metrics in [7, 8] to compute recall scores at different ranks in this task. In the training set, we gradually increase the scale of the dataset. We find that scaling up the dataset from "AudioCaps + Clotho" to "LA." does not improve the result on AudioCaps evaluation set but gets better performance on Clotho evaluation set, which is similar to the comparison between MMT [7] and CLAP-HTSAT [5]. One reason is that AudioCaps contains audios similar to AudioSet on which the audio encoder's loaded checkpoint is pretrained. When the model receives more data from other sources, it increases its generalization but moves the distribution out of AudioSet data. Therefore, the performance on AudioCaps drops but that on Clotho increases a lot, demonstrating a trade-off of the model to keep the performance among different types of audios.

Keyword-to-Caption and Feature Fusion When adding the feature fusion mechanism and keyword-to-caption augmentation to the model, we can observe that either of them improves the performance. The feature fusion is effective especially in Clotho dataset because it contains longer audio data (> 10-second). When we add AudioSet into the training set with either template prompting or keyword-to-caption augmentation, we can see the performance increases again on AudioCaps while decreases on Clotho. This further confirms the trade-off performance between AudioCaps and Clotho datasets mentioned above. And the keyword-to-caption augmentation does bring in better performance than the simple template text prompting method on most metrics.

As the result, our best model outperforms previous methods on most metrics (mainly R@1=36.7% on AudioCaps and R@1=18.2% on Clotho) in the text-to-audio retrieval tasks. We show that training on large-scale datasets (LAION-Audio-630K and AudioSet with keyword-to caption augmentation), and feature fusion can effectively improve model performance.

4.3. Zero-shot and Supervised Audio Classification

Zero-shot Audio Classification To study the model generalization and robustness, we conduct zero-shot audio classification experiments on three top-performing models in previous experiments. We evaluate models on three audio classification dataset, namely ESC-50 [27], VGGSound [28], and Urbansound8K (US8K) [29]. We use **top-1 accuracy** as the metric. We classify audio by performing audio-to-text retrieval with each text corresponds to the text prompt converted from class label via "This a sound of `label`.". We noticed a dataset overlap between our training data and the zero-shot dataset we are evaluating on. We **excluded all the overlap samples** and perform zero-shot evaluation on the whole remaining dataset.

Model	Audio Classification Dataset & Setting				
	ESC-50	US8K	VGGSound		FSD50K
	ZS.	ZS.	ZS.	SV.	SV.
Wav2CLIP [9]	41.4	40.4	10.0	46.6	43.1
AudioClip [3]	69.4	65.3	-	-	-
CLAP [5]	82.6	73.2	-	-	58.6
Ours	89.1	73.2	29.1	**75.4**	64.9
Ours+Fusion	88.0	75.8	26.3	75.3	64.4
Our+K2C Aug.	**91.0**	**77.0**	**46.2**	75.3	59.7
SoTA*	82.6 [5]	73.2 [5]	10.0 [9]	64.1 [25]	**65.6** [26]

Table 4: The zero-shot (ZS.) and supervised (SV.) audio classification results. The SoTA of each dataset/setting is denoted by the reference after the number.

Supervised Audio Classification We perform supervised audio classification by fine-tuning the audio encoder on FSD50K [30] and VGGSounddatasets. We do not conduct this experiment on ESC-50 and Urbansound8K because the potential data leakage issue in those dataset will makes the results incomparable with the previous methods. Specially, **mAP** is used as the metric to evaluate FSD50K.

As shown in the in Table 4, our models achieves new SoTAs of zero-shot audio classification across all three datasets, demonstrating the high generalization ability of our model to unseen data. Keyword-to-Caption augmentation increases the performance of VGGsound and US8K a lot as it adds more text captions to "enrich" the text embedding space. Feature fusion not only enables the model to handle variable-length input, but also achieves better performance than previous models. Our best supervised audio classification result outperforms the current state-of-the-art on VGGSound dataset and is close to state-of-the-art on FSD50K dataset. The results verify that the proposed model also learns efficient audio representation during contrastive learning paradigm.

5. CONCLUSION AND FUTURE WORK

In this paper, we propose a large-scale audio-text dataset and improvements on current language-audio contrastive learning paradigm. We show that LAION-Audio-630, AudioSet with keyword-to-caption augmentation, and feature fusion effectively leads to better audio understanding, task performance, and enables effective learnings on variable-length data. Future works include collecting even larger dataset on training, applying representations into more downstream tasks such as audio synthesis and separation.

6. REFERENCES

[1] Alec Radford, Jong Wook Kim, and Chris Hallacy et al., “Learning transferable visual models from natural language supervision,” *CoRR*, vol. abs/2103.00020, 2021. (Cited on pages 1, 2, and 3.)

[2] Jia Deng, Wei Dong, and Richard Socher et al., “Imagenet: A large-scale hierarchical image database,” in *Proc. CVPR*, 2009. (Cited on page 1.)

[3] Andrey Guzhov, Federico Raue, Jörn Hees, and Andreas Dengel, “Audioclip: Extending clip to image, text and audio,” in *Proc. ICASSP*, 2022. (Cited on pages 1 and 4.)

[4] Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang, “CLAP: learning audio concepts from natural language supervision,” *CoRR*, vol. abs/2206.04769, 2022. (Cited on pages 1 and 4.)

[5] Soham Deshmukh, Benjamin Elizalde, and Huaming Wang, “Audio retrieval with wavtext5k and CLAP training,” *CoRR*, vol. abs/2209.14275, 2022. (Cited on pages 1, 3, and 4.)

[6] Yusong Wu, Tianyu Zhang, and Ke Chen, “Text-to-audio retrieval via large-scale contrastive learning,” *Proc. DCASE*, 2022. (Cited on page 1.)

[7] Andreea-Maria Oncescu, A. Sophia Koepke, and João F. Henriques et al., “Audio retrieval with natural language queries,” in *Proc. Interspeech*, 2021. (Cited on pages 1, 3, 4, and 6.)

[8] Xinhao Mei, Xubo Liu, and Jianyuan Sun et al., “On metric learning for audio-text cross-modal retrieval,” in *Proc. Interspeech*, 2022. (Cited on pages 1, 3, 4, and 6.)

[9] Ho-Hsiang Wu, Prem Seetharaman, Kundan Kumar, and Juan Pablo Bello, “Wav2clip: Learning robust audio representations from clip,” in *Proc. ICASSP*, 2022. (Cited on pages 1 and 4.)

[10] Qiuqiang Kong, Yin Cao, and Turab Iqbal et al., “Panns: Large-scale pretrained audio neural networks for audio pattern recognition,” *IEEE Trans. Audio, Speech, Lang. Process.*, 2020. (Cited on pages 1 and 3.)

[11] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova, “BERT: pre-training of deep bidirectional transformers for language understanding,” in *Proc. NAACL-HLT*, 2019. (Cited on pages 1 and 3.)

[12] Ke Chen, Xingjian Du, Bilei Zhu, Zejun Ma, Taylor Berg-Kirkpatrick, and Shlomo Dubnov, “HTS-AT: A hierarchical token-semantic audio transformer for sound classification and detection,” in *Proc. ICASSP*, 2022. (Cited on pages 1 and 3.)

[13] Yinhan Liu, Myle Ott, and Naman Goyal et al., “Roberta: A robustly optimized BERT pretraining approach,” *CoRR*, vol. abs/1907.11692, 2019. (Cited on pages 1 and 3.)

[14] Jort F. Gemmeke, Daniel P. W. Ellis, and Dylan Freedman et al., “Audio set: An ontology and human-labeled dataset for audio events,” in *Proc. ICASSP*, 2017. (Cited on pages 1 and 2.)

[15] Konstantinos Drossos, Samuel Lipping, and Tuomas Virtanen, “Clotho: an audio captioning dataset,” in *Proc. ICASSP*, 2020. (Cited on page 2.)

[16] A. Sophia Koepke, Andreea-Maria Oncescu, João F. Henriques, Zeynep Akata, and Samuel Albanie, “Audio retrieval with natural language queries: A benchmark study,” *CoRR*, vol. abs/2112.09418, 2021. (Cited on page 2.)

[17] Chris Dongjoo Kim, Byeongchang Kim, Hyunmin Lee, and Gunhee Kim, “Audiocaps: Generating captions for audios in the wild,” in *Proc. NAACL-HLT*, 2019. (Cited on page 2.)

[18] Abien Fred Agarap, “Deep learning using rectified linear units (relu),” *CoRR*, vol. abs/1803.08375, 2018. (Cited on pages 2 and 3.)

[19] Yann LeCun, Léon Bottou, Yoshua Bengio, and Patrick Haffner, “Gradient-based learning applied to document recognition,” *Proc. IEEE*, 1998. (Cited on page 3.)

[20] Ze Liu, Yutong Lin, and Yue Cao et al., “Swin transformer: Hierarchical vision transformer using shifted windows,” in *Proc. ICCV*, 2021. (Cited on page 3.)

[21] Yimian Dai, Fabian Gieseke, and Stefan Oehmcke et al., “Attentional feature fusion,” in *Proc. WACV*, 2021. (Cited on pages 3 and 8.)

[22] Colin Raffel, Noam Shazeer, and Adam Roberts et al., “Exploring the limits of transfer learning with a unified text-to-text transformer,” *JMLR*, 2020. (Cited on pages 3 and 8.)

[23] Diederik P Kingma and Jimmy Ba, “Adam: A method for stochastic optimization,” in *Proc. ICLR*, 2014. (Cited on page 3.)

[24] Priya Goyal, Piotr Dollár, Ross Girshick, Pieter Noordhuis, Lukasz Wesolowski, Aapo Kyrola, Andrew Tulloch, Yangqing Jia, and Kaiming He, “Accurate, large minibatch sgd: Training imagenet in 1 hour,” *arXiv preprint arXiv:1706.02677*, 2017. (Cited on page 3.)

[25] Arsha Nagrani, Shan Yang, Anurag Arnab, Aren Jansen, Cordelia Schmid, and Chen Sun, “Attention bottlenecks for multimodal fusion,” in *Proc. NeurIPS*, 2021. (Cited on page 4.)

[26] Khaled Koutini, Jan Schlüter, Hamid Eghbal-zadeh, and Gerhard Widmer, “Efficient training of audio transformers with patchout,” in *Proc. Interspeech*, 2022. (Cited on page 4.)

[27] Karol J. Piczak, “ESC: dataset for environmental sound classification,” in *Proc. ACM Multimed.*, 2015. (Cited on page 4.)

[28] Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman, “Vggsound: A large-scale audio-visual dataset,” in *Proc. ICASSP*, 2020. (Cited on page 4.)

[29] Justin Salamon, Christopher Jacoby, and Juan Pablo Bello, “A dataset and taxonomy for urban sound research,” in *Proc. ACM Multimed.*, 2014. (Cited on page 4.)

[30] Eduardo Fonseca, Xavier Favory, Jordi Pons, Frederic Font, and Xavier Serra, “FSD50K: an open dataset of human-labeled sound events,” *IEEE Trans. Audio, Speech, Lang. Process.*, 2022. (Cited on page 4.)

[31] Frederic Font Corbera, Gerard Roma Trepat, and Xavier Serra, “Freesound technical demo,” in *Proc. ACM Multimed.*, 2013. (Cited on pages 6 and 7.)

A. APPENDIX

B. ACKNOWLEDGEMENT

Yusong Wu, Ke Chen, Tianyu Zhang are opensource contributors to LAION projects. Our codebase is build on following open-source projects: PANN[4], HTSAT[5], open_clip[6], PyTorch[7]. We would like to thank the support of computation infrastructure from LAION, Stability.ai and Summit cluster from Oak Ridge National Laboratory. This project was proposed in IFT-6167 at University of Monreal and Mila given by Prof. Irina Rish. We would like to thank the support from Christoph Schuhmann, Richard Vencu,, Romain Beaumon, as this project would not be possible without them. We would like to thank the Institute for Research and Coordination in Acoustics and Music (IRCAM)[8] and Project REACH[9]: Raising Co-creativity in Cyber-Human Musicianship for supporting this project. We would like to thank all the community contributors for contributing the collection of LAION-630k dataset. Those community contributors (Discord ids) include but not limited to: @marianna13#7139, @Chr0my#0173, @PiEquals4#1909, @Yuchen Hui#8574, @Antonioooooo#4758, @IYWO#9072, krishna#1648, @dicknascarsixtynine#3885, and @turian#1607. We would like to thank Xinhao Mei for explaining and helping on retrieval metrics.

C. DETAILS OF EVALUATING RETRIEVAL PERFORMANCE

In this study, the primary focus is on assessing the efficacy of the models in terms of retrieval performance, utilizing metrics such as R@1, R@5, R@10 and Mean Average Precision (mAP). The Clotho and AudioCaps datasets, in particular, are characterized by the presence of five text ground-truths per audio sample. Therefore, in evaluating the retrieval performance on these datasets, we adopt the same metrics as used in previous studies, specifically, those outlined in [7, 8][10].

For text-to-audio retrieval, we treat each text from an audio as independent test sample, and calculate the average of text-to-audio retrieval metrics on test samples that are five times the size of test set. In evaluating audio-to-text recall, the recall for each audio is calculated by taking the best audio-to-text retrieval result from the five text ground-truths. Additionally, audio-to-text Mean Average Precision (mAP) is calculated as $mAP@10 = \frac{1}{R}\sum_{r=1}^{10}(P(r) * rel(r))$, where $P(r)$ represents the precision at recall level r, and $rel(r)$ is a binary indicator of whether the text at recall level r is relevant or not.

In the case of other datasets, such as Freesound, in which there is only one text associated with each audio sample, the recall and mean average precision (mAP) are measured in the standard manner.

D. DETAILS OF LAION-AUDIO-630K

Regarding the section 2.1 and section 2.2 of the paper:

- We list the specifications of website/sources from which we collect the audio samples and text captions for LAION-Audio-630K in Table 5.
- We list the details of three datasets in Table 6. We use the combination of them to train the model in the section 4 of the submission.
- Regarding the section 3.4 of the paper, we present the distribution of audio length on Epidemic Sound and Freesound [31], as parts of LAION-Audio-630K, to demonstrate the existence of variable-length problem in audio data processing and model training.

Data Source	Number of Samples	Duration	Data Type
BBC sound effects	15973	463.48hrs	1 caption per audio, audio
Free To Use Sounds	6370	175.73hrs	Filename as caption, audio
Sonniss Game effects	5049	84.6hrs	Filename as caption, audio
We Sound Effects	488	12.00hrs	Filename as caption, audio
Paramount Motion Sound Effects	4420	19.49hrs	Filename as caption, audio
Audiostock	10000	46.30hrs	1 caption per audio, audio
Freesound [31]	515581	3003.38rs	1-2 captions per audio, audio
Epidemic Sound	75645	220.41hrs	2 captions per audio, audio

Table 5: LAION-Audio-630k Datasets

[4] https://github.com/qiuqiangkong/audioset_tagging_cnn

[5] https://github.com/RetroCirce/HTS-Audio-Transformer

[6] https://github.com/mlfoundations/open_clip

[7] https://pytorch.org/

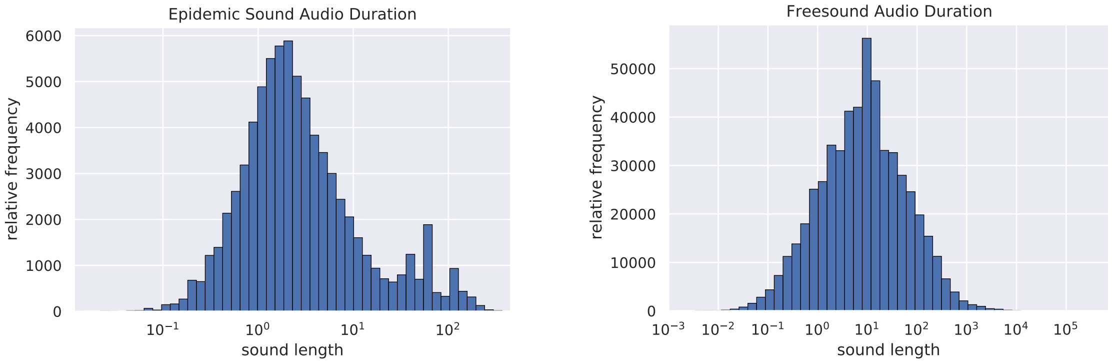
[8] https://www.ircam.fr/

[9] https://www.ircam.fr/projects/pages/reach-project

[10] We implemented the exact evaluation metric in https://github.com/XinhaoMei/audio-text_retrieval/blob/main/tools/utils.py#L74

Data Source	Number of Samples	Duration	Data Type
AudioCaps + Clotho			
AudioCaps	49274	136.87hrs	1 caption per audio, audio
Clotho	3839	23.99hrs	5 captions per audio, audio
LAION-Audio-630K			
BBC sound effects	15973	463.48hrs	1 caption per audio, audio
Episodesound	75645	220.41hrs	2 captions per audio, audio
freesound	414127	2528.15hrs	1-2 captions per audio, audio
Free To Use Sounds	6370	175.73hrs	Filename as caption, audio
Sonniss Game effects	5049	84.6hrs	Filename as caption, audio
We Sound Effects	488	12.00hrs	Filename as caption, audio
Paramount Motion Sound Effects	4420	19.49hrs	Filename as caption, audio
Audiostock	10000	46.30hrs	1 caption per audio, audio
FSD50K	36796	70.39hrs	1 caption per audio, audio
MACS	3537	9.825hrs	Several (2∼) captions per audio, audio
Wavtext5K	4072	23.2hrs	1 caption per audio, audio
AudioSet			
AudioSet	1912024	463.48hrs	2 captions per audio,audio

Table 6: Training Datasets



(a) The audio length distribution of Epidemic Sound

(b) The audio length distribution of Freesound

Fig. 2: The audio length distribution of Epidemic Sound and Freesound.

D.1. Freesound Dataset

The samples in Freesound dataset are collected from Freesound [31]. All audio clips from Freesound are released under Creative Commons (CC) licenses, while each clip has its own license as defined by the clip uploader in Freesound, some of them requiring attribution to their original authors and some forbidding further commercial reuse. Specifically, here is the statistics about licenses of audio clips involved in LAION-Audio-630K:

- CC-BY: 196884
- CC-BY-NC: 63693
- CC0: 270843
- CC Sampling+: 11556

We listed the licenses for each sample in our dataset release page[11].

E. ATTENTIONAL FEATURE FUSION

Regarding the section 3.4 of the paper, we demonstrate the “attentional feature fusion” architecture, a two-branch CNN network, to show how we combine the global information and the local information of input audios together.

As shown in Figure 3, the fusion architecture accepts two inputs: X is the global information (X^{a}_{global}), and Y is the merged local information (X^{a}_{local}) Two inputs are sent to two CNN networks to generate the coefficient, then X and Y are added by this coefficient.

[11] `https://github.com/LAION-AI/audio-dataset/tree/main/laion-audio-630k`

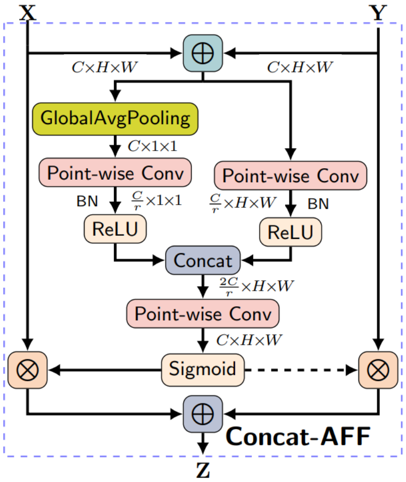


Fig. 3: The attentional feature fusion architecture from [21].

F. ADDITIONAL EXPERIMENT OF FEATURE FUSION ON FREESOUND DATASET

Regarding to the section 4.2 of the paper, to further evaluate the efficacy of feature fusion, apart from AudioCaps and Clotho datasets, we further evaluate our model on Freesound evaluation set, which contains more than 10-sec audio samples (similar to Clotho dataset).

The result is shown in Table 7, the notation is the same as the Table 3 in our submission paper. The performance on Freesound dataset shares a similar trend with that on Clotho dataset:

- the performance trained on "AudioCaps + Clotho + LA." is better than that trained on "AudioCaps + Clotho + LA. + AudioSet". As demonstrate in the section 4.2, similar to Clotho, the Freesound dataset contains audio samples that are different from AudioSet, adding the AudioSet into the training will move the model's distribution out of general audio data to AudioSet-like audio data, such decreasing the performance.
- the performance with feature fusion is better than that without feature fusion, as the Freesound dataset contains the samples larger than 10-secs, which is the same to Clotho dataset. Their performance trend are similar.

From the above experiment, we can further conclude that the feature fusion can improve the performance of text-to-audio task (i.e., generate better audio representations) on the variable-length audio samples.

Model	Training Set	Freesound (mAP@10)	
		A→T	T→A
HTSAT-RoBERTa	AudioCaps + Clotho + LA.	25.9	24.5
HTSAT-RoBERTa (fusion)	AudioCaps + Clotho + LA.	**26.4**	**24.9**
HTSAT-RoBERTa	AudioCaps + Clotho + LA. + AudioSet (K2C Aug.)	22.9	21.8
HTSAT-RoBERTa (fusion)	AudioCaps + Clotho + LA. + AudioSet (K2C Aug.)	24.6	22.9

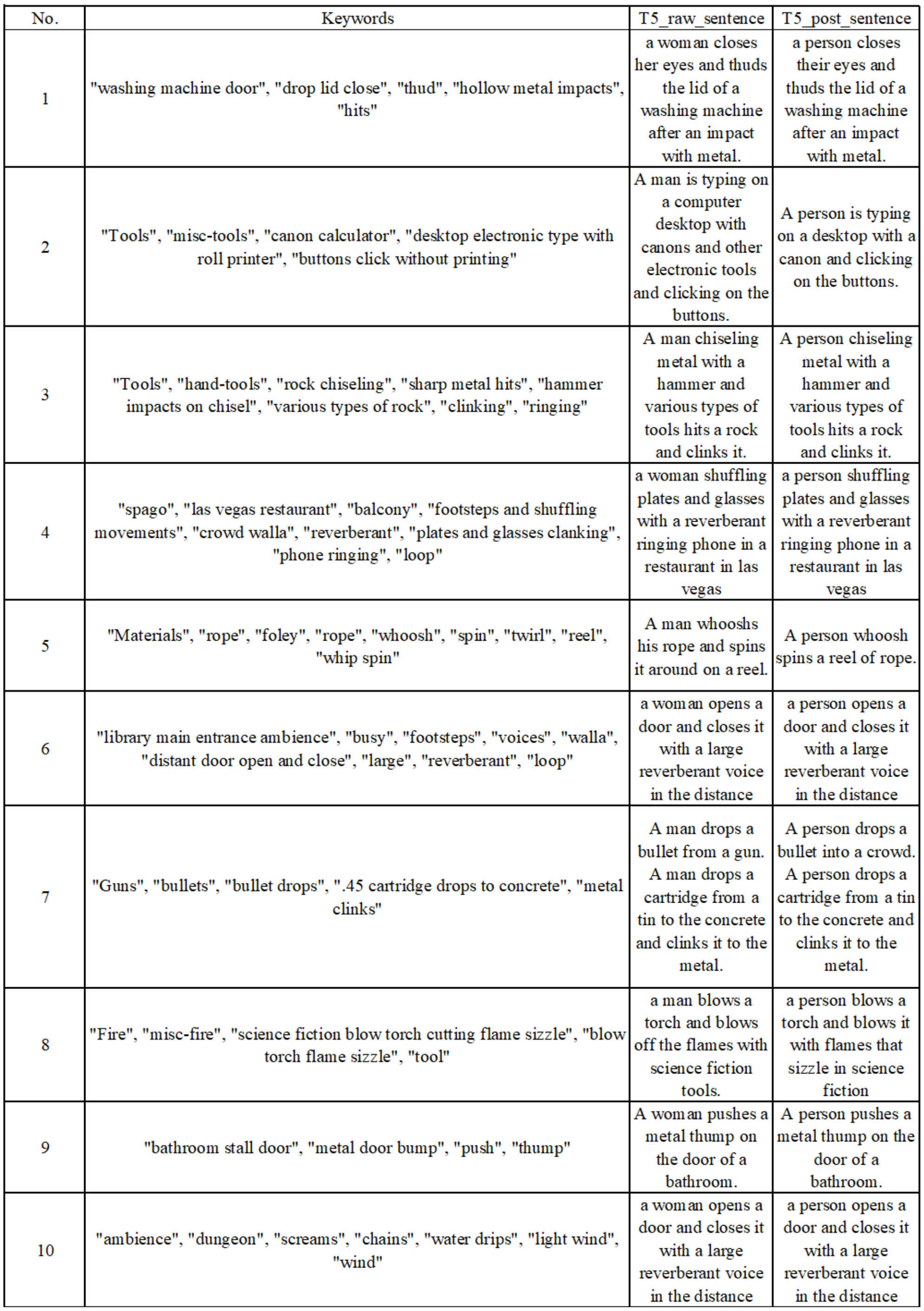
Table 7: The text-to-audio retrieval performance on Freesound evaluation set.

G. EXAMPLES OF KEYWORD-TO-CAPTION AUGMENTATION

Regarding the section 3.5 of the paper, we show some examples of keyword-to-caption by T5 model [22][12] from AudioSet labels in the below Table 4. And the de-biased version for the model training.

Additionally, when applying keyword to caption, we excluded samples shorter than 2 seconds, as we found in such case the audio is merely a single event, thus matching poorly with the caption generated. When using keyword to caption in training dataset including audioset, we use only the captions generated by keyword to caption and exclude the captions generated by template.

[12] We use the T5 model provided in `https://github.com/gagan3012/keytotext`.



No.	Keywords	T5_raw_sentence	T5_post_sentence
1	"washing machine door", "drop lid close", "thud", "hollow metal impacts", "hits"	a woman closes her eyes and thuds the lid of a washing machine after an impact with metal.	a person closes their eyes and thuds the lid of a washing machine after an impact with metal.
2	"Tools", "misc-tools", "canon calculator", "desktop electronic type with roll printer", "buttons click without printing"	A man is typing on a computer desktop with canons and other electronic tools and clicking on the buttons.	A person is typing on a desktop with a canon and clicking on the buttons.
3	"Tools", "hand-tools", "rock chiseling", "sharp metal hits", "hammer impacts on chisel", "various types of rock", "clinking", "ringing"	A man chiseling metal with a hammer and various types of tools hits a rock and clinks it.	A person chiseling metal with a hammer and various types of tools hits a rock and clinks it.
4	"spago", "las vegas restaurant", "balcony", "footsteps and shuffling movements", "crowd walla", "reverberant", "plates and glasses clanking", "phone ringing", "loop"	a woman shuffling plates and glasses with a reverberant ringing phone in a restaurant in las vegas	a person shuffling plates and glasses with a reverberant ringing phone in a restaurant in las vegas
5	"Materials", "rope", "foley", "rope", "whoosh", "spin", "twirl", "reel", "whip spin"	A man whooshs his rope and spins it around on a reel.	A person whoosh spins a reel of rope.
6	"library main entrance ambience", "busy", "footsteps", "voices", "walla", "distant door open and close", "large", "reverberant", "loop"	a woman opens a door and closes it with a large reverberant voice in the distance	a person opens a door and closes it with a large reverberant voice in the distance
7	"Guns", "bullets", "bullet drops", ".45 cartridge drops to concrete", "metal clinks"	A man drops a bullet from a gun. A man drops a cartridge from a tin to the concrete and clinks it to the metal.	A person drops a bullet into a crowd. A person drops a cartridge from a tin to the concrete and clinks it to the metal.
8	"Fire", "misc-fire", "science fiction blow torch cutting flame sizzle", "blow torch flame sizzle", "tool"	a man blows a torch and blows off the flames with science fiction tools.	a person blows a torch and blows it with flames that sizzle in science fiction
9	"bathroom stall door", "metal door bump", "push", "thump"	A woman pushes a metal thump on the door of a bathroom.	A person pushes a metal thump on the door of a bathroom.
10	"ambience", "dungeon", "screams", "chains", "water drips", "light wind", "wind"	a woman opens a door and closes it with a large reverberant voice in the distance	a person opens a door and closes it with a large reverberant voice in the distance

Fig. 4: Examples of keyword-to-caption augmentation from AudioSet labels and the de-biased version for the model training.

H. EXPERIMENT SETTINGS ON DATA EXCLUSION

Regarding the section 4.3 of the paper, we excluded all the overlap samples and perform zero-shot evaluation on the whole remaining dataset. The below table 8 shows the detail of it.

Datasource A	Datasource B	Number of samples from Datasource A that are also in Datasource B
ESC50-all	Clotho-train	94
ESC50-all	Clotho-valid	27
ESC50-all	Clotho-test	34
ESC50-all	FSD50K-train	399
ESC50-all	FSD50K-valid	60
ESC50-all	FSD50K-test	171
USD8K-all	Clotho-train	411
USD8K-all	Clotho-valid	150
USD8K-all	Clotho-test	209
USD8K-all	FSD50K-train	697
USD8K-all	FSD50K-valid	180
USD8K-all	FSD50K-test	341
Clotho-test	FSD50K-train	54
Clotho-test	FSD50K-valid	15
Clotho-test	FSD50K-test	33
FSD50K-test	Clotho-train	137
FSD50K-test	Clotho-valid	31
FSD50K-test	Clotho-test	33
Clotho-valid	FSD50K-train	53
Clotho-valid	FSD50K-valid	10
FSD50K-valid	Clotho-train	38
FSD50K-valid	Clotho-valid	10
Audiocaps-test	Audioset-unbalanced-train	4875
Audiocaps-test	Audioset-balanced-train	0
audioset-test	audiocaps-train	0
audioset-test	audiocaps-valid	0

Table 8: The overlaps between the training data and the zero-shot evaluation data, we excluded all these overlaps from the evalation sets to calculate the audio classification metrics.