PRE-TRAINED MODELS, DATASETS, DATA AUGMENTATION FOR LANGUAGE-BASED AUDIO RETRIEVAL

Hokuto Munakata, Taichi Nishimura, Shota Nakada, Tatsuya Komatsu

LY Corporation, Japan

ABSTRACT

We investigate the impact of pre-trained models, datasets, and data augmentation on language-based audio retrieval. Despite the high interest in cross-modal retrieval and the introduction of various datasets, powerful encoders, and data augmentation techniques, it remains unclear which approaches are most effective for language-based audio retrieval. We focus on which should be selected to build a retrieval model. First, we investigate the performance gain by four audio encoders, PaSST, CAV-MAE, BEATs, and VAST, and three text encoders BERT, RoBERTa, and T5. Second, we prepare massive datasets of over 670k audio-text pairs including ClothoV2, AudioCaps, WavCaps, MACS, and Auto-ACD. Third, we investigate the combination of data augmentation methods to enhance the retrieval performance including mixup-contrast and text token masking. In addition, we also explore inference time augmentation by paraphrasing textual queries using Chat-GPT to achieve robust retrieval performance. Our final results achieve 39.79 points with a single model and 42.22 points with the ensemble models in the mean average precision among the top 10 results on the evaluation split of ClothoV2.

Index Terms— Language-based audio retrieval, Pre-trained model, Data augmentation,

1. INTRODUCTION

Language-based audio retrieval systems take a textual query as input and retrieve the corresponding audio from a database. The mainstream approach projects both audio and text data into a joint embedding space calculates their similarity, and ranks the audio based on this similarity [1] (Figure 1). To obtain this joint space, models are trained from audio-text pairs. Contrastive learning is a dominant training method, where positive audio-text pairs are given higher similarity scores and negative pairs lower scores [2].

This task shares common characteristics with text-to-image/video retrieval because both tasks involve processing language inputs and employ contrastive learning to train the dual encoders. In text-to-image/video retrieval, researchers have explored various encoders (e.g., CLIP [3], VideoBERT [4], BERT [5], RoBERTa [6]) trained on massive datasets with data augmentation methods (e.g., Mixco [7]). As with the visual domain, in the audio domain, the encoders, datasets, and data augmentation have been proposed [8, 9]. However, it remains unclear which approaches are most effective for audio retrieval. This motivates us to focus on which should be selected to train the retrieval model.

To address this, we investigate the impact of pre-trained models, datasets, and data augmentation on language-based audio retrieval. For pre-trained models, we use five audio encoders and three text encoders that have achieved state-of-the-art performance in downstream tasks. Specifically, we adopt PaSST [10], BEATs [11], CAV-MAE [12], and VAST [13] for the audio encoder and BERT [5], RoBERTa [6], and T5 [14]. For datasets, we prepare a massive dataset containing over 670k audio-text pairs. The dataset includes both manually annotated data such as ClothoV2 [1], AudioCaps [15], MACS [16], and WavCaps/Auto-ACD[17, 18] that utilize large language models (LLMs) to generate pseudo audio-text pairs. For data augmentation, we apply multiple data augmentation methods, mixup-contrast [7], and text token masking for further improvement of the retrieval performance. In addition, we also explore inference time augmentation by paraphrasing textual queries using Chat-GPT to achieve robust retrieval performance.

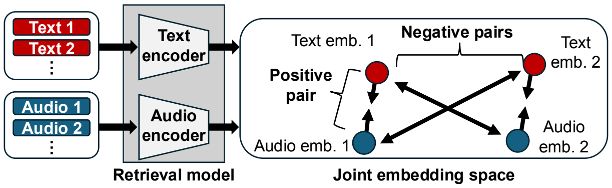


Figure 1: An overview of the conventional language-based audio retrieval system based on contrastive learning. Through contrastive learning, positive pairs of audio-text embeddings have similar values, while negative pairs have less similar values.

In our experiments, we conduct thorough comparative studies on encoders, datasets, data augmentation, and inference time augmentation. As a result, we provide the following three insights. First, in terms of encoders, VAST and RoBERTa yield the best performance. The performance of audio encoders aligned with the performance in the AudioSet classification task except for PaSST, which adopts patch-out, a regularization technique. Second, for the datasets, we observe that it is important to improve the text annotation quality, rather than increase the pseudo audio-text pairs generated from LLMs. Third, data augmentation approaches, both training and inference augmentation, contribute to the performance gain. As a result of combining these techniques, the model with PaSST and RoBERTa yields the best performance, achieving 39.79 mean average precision (mAP) on the ClothoV2 evaluation split. An ensemble of multiple models reaches 42.26 mAP.

2. MODEL OVERVIEW

As with recent cross-modal retrieval, our training approach is based on contrastive learning. The retrieval model has audio and text encoders to project the audio and text onto the joint embedding space. The input audio $\mathbf{X}^{(A)}$ and text $\mathbf{X}^{(T)}$ is projected onto D-dimensional joint space as $\mathbf{Z}^{(A)}$ and $\mathbf{Z}^{(T)}$ by audio/text encoders f_A and f_T as follows:

$$\mathbf{Z}^{(A)} = f_A(\mathbf{X}^{(A)}), \tag{1}$$

$$\mathbf{Z}^{(T)} = f_T(\mathbf{X}^{(T)}). \tag{2}$$

Based on $(\mathbf{Z}^{(A)}, \mathbf{Z}^{(T)})$, the model is trained to discriminate positive and negative from each pair of B audio and text samples based on InfoNCE [2] loss. Specifically, let i-th audio and j-th text be $\mathbf{Z}_i^{(A)}$ and $\mathbf{Z}_j^{(T)}$, where $i = j$ is positive and $i \neq j$ is negative. The loss is written as the cross-entropy loss with the softmax as follows:

$$\mathcal{L}_{\text{CE}}\left(\mathbf{Z}, \mathbf{Z}'_k, \mathbf{Z}'_:\right) = -\log \frac{\exp\left(S(\mathbf{Z}, \mathbf{Z}'_k)/\tau\right)}{\sum_{\mathbf{Z}' \in \mathbf{Z}'_:} \exp\left(S(\mathbf{Z}, \mathbf{Z}')/\tau\right)}, \quad (3)$$

$$\mathcal{L}_{\text{infoNCE}}^{A \to T} = \sum_i \mathcal{L}_{\text{CE}}\left(\mathbf{Z}_i^{(A)}, \mathbf{Z}_i^{(T)}, \mathbf{Z}_:^{(T)}\right), \quad (4)$$

$$\mathcal{L}_{\text{infoNCE}}^{T \to A} = \sum_j \mathcal{L}_{\text{CE}}\left(\mathbf{Z}_j^{(T)}, \mathbf{Z}_j^{(A)}, \mathbf{Z}_:^{(A)}\right), \quad (5)$$

$$\mathcal{L}_{\text{infoNCE}} = \mathcal{L}_{\text{infoNCE}}^{A \to T} + \mathcal{L}_{\text{infoNCE}}^{T \to A}, \quad (6)$$

where $\mathbf{Z}'_:$ is the set of $\mathbf{Z}'_1, ..., \mathbf{Z}'_B$, S is the cosine similarity, and τ is a trainable temperature parameter.

In the inference stage, the textual query and audio in the database are projected onto the joint embedding space, and their similarity is calculated to rank the audio. The audio ranked at position k in the database $\mathbf{X}_k^{(A)}$ is obtained by sorting the cosine similarities as follows:

$$\mathbf{X}_k^{(A)} = \underset{\mathbf{X}^{(A)} \in \mathbf{X}_{\text{DB}}^{(A)}}{\text{argmax}_k} \; S(f_A(\mathbf{X}^{(A)}), f_T(\mathbf{X}^{(T)})), \quad (7)$$

where argmax_k is the operation of extracting the k-th largest element and $\mathbf{X}_{\text{DB}}^{(A)}$ is the set of audio in the database.

3. AUDIO AND TEXT ENCODERS

3.1. Audio Encoder

The first focus of this study is encoders. For the audio side, we investigate four audio encoders: PaSST [10], BEATs [11], VAST [13], and CAV-MAE [12]. These models are variants of ASTs [19] that apply Vision Transformers [20] to audio spectra.

PaSST [10] enhances the AST by incorporating the patch-out technique that improves the generalization and accelerates the training by dropping the parts of the input sequence. Additionally, it employs distinct positional encodings for the time and frequency dimensions, leading to performance enhancements. We used the weights pre-trained on the AudioSet classification task. The stride size for the frequency and time was 16 and the patches were not overlapped. Only for this model, we apply patch-out [10] with 2 and 15 patches for the frequency and time directions during training.

CAV-MAE [12] extends the AST into an audio-visual model by integrating the outputs of AST and Vision Transformer [21]. This combined output is fed into a subsequent transformer that captures the interrelationships between audio and visual modalities through self-attention mechanisms. A multi-task loss that combines contrastive learning and masked autoencoder loss on both AudioSet and VGGSound datasets is used in the training. We used the scale++ model.

BEATs [11] introduces a discrete audio tokenizer to the AST framework, leveraging SSL. AST-based SSL model and the audio tokenizer are trained alternately in a repeated manner. Notably, BEATs demonstrated its high performance by being employed in the best system for the audio captioning task of the DCASE 2023 Challenge. We used the weights fine-tuned on the AudioSet classification task.

VAST [13] is a multi-modal model that integrates vision, audio, and texts into a unified framework using BEATs for the audio encoder. It is trained on the VAST-27M dataset, which includes 27 million video clips with vision-text or audio-text. The model trained with the dataset for various tasks such as retrieval, captioning, and question answering. VAST has demonstrated state-of-the-art performance on multiple cross-modality benchmarks. We used two different weights only pre-trained based on SSL and fine-tuned for the audio captioning task.

3.2. Text Encoder

For the text side, we investigate three text encoders: BERT [5], RoBERTa [6], T5 [14]. These models are based on Transformer architecture [22] trained with large-scale crawled text corpora. We use pre-trained weights of the large model of these encoders publicly available on HuggingFace.

BERT [5] is the bidirectional transformers encoder to improve understanding ability of the context of words in a sentence. This model is pre-trained based on masked language modeling (MLM) and next sentence prediction (NSP) with English Wikipedia and BookCorpus [23] containing over 3500 million words. We used the large model.

RoBERTa [6] is an optimized version of BERT pre-trained with diverse corpora of 160 GB. It removes NSP and focuses solely on MLM objectives. We used the large model.

T5 [14] is the transformer-based encoder and decoder architecture, which formulates a wide range of tasks such as sentence prediction. This model is trained based on SSL using multiple objective functions with C4 dataset [14], a web-crawled corpus of about 750 GB.

4. DATASET AND AUGMENTATION

4.1. Datasets

The second focus is the dataset. We prepare six datasets to investigate which one contributes to the retrieval performance: ClothoV2 [1], ClothoV2-GPT [24], MACS [16], AudioCaps [15], WavCaps [17], and Auto-ACD [18]. Note that all texts are preprocessed by removing punctuation and converting it to lowercase. In our evaluation, we use the ClothoV2 evaluation split and AudioCaps test split.

ClothoV2 [1] contains audio recordings ranging from ten to 30 seconds in length. The dataset is divided into training, validation, and test splits with 3840, 1045, and 1043 recordings, respectively. Each audio recording in the dataset is associated with five human-written captions containing eight to 12 words.

ClothoV2-GPT [24] is an augmented version of Clotho v2, where the original manually annotated text is expanded by five additional texts generated by OpenAI's GPT3.5-turbo. Five additional captions are generated by GPT based on the original audio's captions and keywords from metadata.

MACS [16] is extracted from the TAU Urban Acoustic Scenes 2019 and contains approximately 3,900 samples, each ten seconds long, totaling around 47 hours of audio. The captions are manually created, with roughly five captions per audio clip. The vocabulary size is 2803 words.

AudioCaps [15] is created by manually annotating a subset of The available subset of the dataset divided into training, validation, and test splits with 46163, 457, and 911 recordings, respectively. Most of the clips are 10 seconds long. The captions are manually created, with one caption per audio clip. The vocabulary size is 5129 words.

WavCaps [17] includes samples from FreeSound, BBC Sound Effects, SoundBible, and AudioSetSL. It contains around 400k samples in total. The clip lengths vary from ten seconds to several minutes, with an average length of 67 seconds, totaling approximately 7500 hours of audio. Captions are automatically generated using GPT based on existing metadata (tags, etc.) and different prompts are used for each source dataset. Each audio clip has one caption, with a vocabulary size of 28721 words.

Auto-ACD [18] comprises samples from AudioSet and VGGSound [25]. We used the subset from VGGSound because it performs better on Clotho. It contains 180k samples generated from the YouTube video data of VGGSound. The text was generated by OpenAI's GPT leveraging existing tags and object recognition results from videos. Most clips are 10 seconds long, totaling approximately 500 hours of audio. Each audio clip has one caption, with a vocabulary size of 8157 words.

4.2. Training Data Augmentation

The third focus is the data augmentation. We use the following two approaches: Mix-up contrast (Mixco) [7] and text token masking.

Mixco [7] is a data augmentation method for contrastive learning. It was originally used for text-to-image retrieval and achieved significant performance gain. Mixco introduces the semi-positive pair, which is the pair of an image generated by mixing two images and their corresponding texts. To enable the model to learn better representations, the target labels for semi-positive pairs in the cross-entropy loss are set as soft labels rather than hard ones. To apply Mixco to language-based audio retrieval, we mix the i-th audio in the batch $\mathbf{X}_i^{(A)}$ and another audio $\mathbf{X}_{\phi(i)}^{(A)}$ in the waveform and transform it as follows:

$$\mathbf{X}_i^{(A')} = \lambda \mathbf{X}_i^{(A)} + (1-\lambda)\mathbf{X}_{\phi(i)}^{(A)}, \tag{8}$$

$$\mathbf{Z}_i^{(A')} = \mathsf{AudioEncoder}(\mathbf{X}_i^{(A')}), \tag{9}$$

where $\phi(i)$ is a randomly selected index for i and $\lambda \in (0,1)$ is a random variable sampled from the uniform distribution. From the embeddings of the mixtures, the additional loss of Mixco is obtained by the weighted sum of the infoNCE loss to discriminate semi-positive and negative pairs similar to Eq. (4) and Eq. (5) as follows:

$$\begin{aligned}\mathcal{L}_{\text{mixco}}^{A\to T} = \sum_i \lambda \Big\{ & \mathcal{L}_{\text{CE}}\left(\mathbf{Z}_i^{(A')}, \mathbf{Z}_i^{(T)}, \mathbf{Z}_:^{(T)}\right) \\ & +(1-\lambda)\ \mathcal{L}_{\text{CE}}\left(\mathbf{Z}_i^{(A')}, \mathbf{Z}_{\phi(i)}^{(T)}, \mathbf{Z}_:^{(T)}\right)\Big\}, \end{aligned} \tag{10}$$

$$\begin{aligned}\mathcal{L}_{\text{mixco}}^{T\to A} = \sum_j \Big\{ & \lambda \mathcal{L}_{\text{CE}}\left(\mathbf{Z}_j^{(T)}, \mathbf{Z}_j^{(A')}, \mathbf{Z}_:^{(A')}\right) \\ & +(1-\lambda)\ \mathcal{L}_{\text{CE}}\left(\mathbf{Z}_{\phi(j)}^{(T)}, \mathbf{Z}_j^{(A')}, \mathbf{Z}_:^{(A')}\right)\Big\}, \end{aligned} \tag{11}$$

$$\mathcal{L}_{\text{mixco}} = \mathcal{L}_{\text{mixco}}^{A\to T} + \mathcal{L}_{\text{mixco}}^{T\to A}. \tag{12}$$

We use the same temperature parameter for Eq. (3). In our experiment, we use the combination of the original info NCE loss and Mixco loss: $\mathcal{L} = \mathcal{L}_{\text{infoNCE}} + \mathcal{L}_{\text{mixco}}$.

Text token masking is a data augmentation method for the input text to mitigate overfitting. The text tokens are randomly replaced with `[MASK]` token for BERT and RoBERTa, and `<extra_id_0>` for T5. We set the replace probability to 15%.

Table 1: Performance by the audio and text encoders. The columns for mAP@10 represent the average and standard deviation achieved by the three models

ID	Audio encoder	Text encoder	mAP@10	
			ClothoV2	AudioCaps
A	PaSST	RoBERTa	39.77 ± .07	52.45 ± .32
B	CAV-MAE	RoBERTa	38.57 ± .77	51.52 ± .95
C	BEATs	RoBERTa	39.25 ± .14	54.70 ± .10
D	VAST (captioning)	RoBERTa	39.68 ± .09	**55.49 ± .06**
E	VAST (vanilla)	RoBERTa	**39.79 ± .14**	55.22 ± .31
F	PaSST	T5	36.06 ± .10	50.76 ± .23
G	PaSST	BERT	36.27 ± .20	49.03 ± .16

Table 2: Performance by the training dataset. The second column represents the number of audio-text pairs. The third column represents how the text data was created.

Training datasets	# of samples	mAP@10	
		ClothoV2	AudioCaps
1. ClothoV2	19k	27.30 ± .43	23.16 ± .28
2. ClothoV2-GPT	19k	27.36 ± .55	24.75 ± .51
3. AudioCaps	46k	23.59 ± .25	49.28 ± .18
4. WavCaps	401k	34.14 ± .38	44.22 ± .47
5. MACS	17k	8.30 ± .41	9.59 ± .42
6. Auto-ACD	185k	21.79 ± .26	28.82 ± .36
1 & 3 & 4	473k	38.40 ± .24	50.81 ± .36
2 & 3 & 4	473k	38.21 ± .09	51.05 ± .21
1 & 3 & 4 & 5	484k	38.80 ± .34	51.30 ± .07
1 & 3 & 4 & 6	651k	38.88 ± .33	51.92 ± .14
1 & 3 & 4 & 5 & 6	670k	**39.09 ± .43**	**52.18 ± .17**

4.3. Inference Time Augmentation

In addition to the training data augmentation, we also devise an inference time query augmentation method by paraphrasing textual queries using Chat-GPT to achieve robust retrieval performance. For example, a query of "A man walking who is blowing his nose hard and about to sneeze." is paraphrased to "A man walks while blowing his nose loudly" and "A man blows his nose hard as he walks." Since ClothoV2-GPT is generated only for the training split, we generated the same format dataset of the ClothoV2 evaluation split and AudioCaps test split using the same prompt of [24] except for not using the keywords. The text encoder projects the original and the additional queries and then the embeddings of each query are averaged.

5. EXPERIMENT

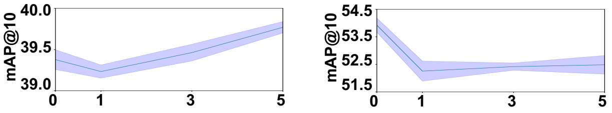
We conduct five experiments to confirm the effect of the encoders, datasets, training data augmentation, inference time augmentation, and the model ensemble.

5.1. Experimental Setting

The dimension of the joint embedding space is set to 1024. The number of training epochs and batch size are 15 and 128, respectively. The optimizer is AdamW [26]. The learning rate was changed by iterations using a cosine scheduler with 1 warm-up epoch and the maximum learning rate was 1×10^{-5}. The initial value of the temperature parameter τ used in Eq (3) was 0.02.

Table 3: Performance with and without data augmentation.

Mixco	Text token masking	mAP@10 ClothoV2	mAP@10 AudioCaps
-	-	39.09 ± .43	52.18 ± .17
✓	-	39.01 ± .27	49.25 ± .89
-	✓	39.33 ± .10	**52.83** ± .22
✓	✓	**39.77 ± .07**	52.45 ± .32



(a) ClothoV2 (b) AudioCaps

Figure 2: The relationship between the retrieval performance and the number of additional queries. The lines and areas represent the Average and standard deviation of mAP@10, respectively.

To avoid learning unexpected relationships between the audio and text caused by the difference among the datasets, we generate each batch from the same dataset. In the training, we conducted validation by 20% of each epoch and saved the model weight. After training, the weights of the models that achieved the top 10 in validation mAP@10 of ClothoV2 are averaged to form the final model weights. For the inference time augmentation, we generate five additional captions for ClothoV2 and the evaluation dataset of this challenge. The replacement probability of the original caption to the generated caption of ClothoV2-GPT is set to be 0.3 as with [24]. The preprocess and sampling rate of the audio follows the original implementation of each audio encoder. Audio clips are trimmed to 10 seconds if they are longer and padded if they are shorter. To mitigate performance variations due to initialization, we train all models three times. We performed the model ensemble by averaging the cosine similarity calculated by each model. In addition, to increase the diversity of the ensemble without additional training, two embeddings are obtained from a single model with or without inference time augmentation.

5.2. Results

Which encoders are the best? Table 1 shows the performance when changing the audio and text encoders. Note that in this experiment, we use all of the datasets, data augmentation, and inference time augmentation. We observe that RoBERTa achieves the best performance on both datasets and the two weights of VAST achieved the best performance on each dataset. In terms of the audio encoder, the performance in the AudioCaps is aligned with the performance in the AudioSet classification task. However, the performance in ClothoV2 is not aligned and PaSST is comparable to VAST. An important difference between PaSST and the other encoders is patch-out, encouraging PaSST to avoid overfitting. Among the text encoders, the performance of T5 is significantly worse than other models. This suggests that bi-directional text encoders (e.g., BERT and RoBERTa) are desirable for the language-based audio retrieval, rather than the uni-directional model (e.g., T5).

Which datasets have a significant impact on performance? Table 2 shows the performance when changing the training dataset. Note that in this experiment, we do not use data augmentation, and the audio/text encoders are PaSST and RoBERTa, respectively. When comparing models trained on ClothoV2 and ClothoV2-GPT, there was no significant performance difference, whether they were used as a single dataset (rows 1 and 2) or as subsets of multiple datasets (rows 7 and 8). The model trained only with MACS and Auto-ACD did not perform well (rows 5 and 6). In contrast, the model with WavCaps shows high performance for both evaluation datasets (row 4). When comparing the improvement of MACS and Auto-ACD (rows 8 and 9), despite the large difference in the number of samples, the difference in the performance was lower than 0.1 point for ClothoV2. When comparing Auto-ACD and WavCaps, the performance gap can be attributed to the fact that Auto-ACD does not implement the multiple filtering processes used by WavCaps. This suggests that acquiring high-quality text is crucial for training effective audio retrieval models.

Table 4: Performance of the ensembles of the multiple models

Model	mAP@10 ClothoV2	mAP@10 DCASE eval.
Ensemble of A, B, C, D, E	42.22	39.2
Our system of DCASE 2024	**42.26**	38.8
The best system of DCASE 2024	41.90	**41.6**
The best system of DCASE 2023	41.42	40.1

Which training data augmentation is the best? Third, we analyze the effect of the data augmentation and the summary is described in Table 3. In this experiment, we used all datasets, and the audio/text encoders are PaSST and RoBERTa. We obtain two findings. First, we separately conduct experiments on text token masking and Mixco and observe that text token masking slightly improves the performance yet Mixco does not. This may be because Mixco does not add new training text patterns, leading to the model's overfitting. Second, the combination of Mixco and text token masking significantly improves the performance. This result indicates that text token masking prevents the model from overfitting, enabling Mixco to be effective.

How many queries are necessary for inference time augmentation? Figure 2a and 2b show the performance change when varying the number of additional queries on ClothoV2 and AudioCaps. The results suggest that the number of additional queries depends on the datasets. In ClothoV2, five additional queries achieve the highest performance, whereas only one additional query has a negative impact. Based on these, we can say that the additional query supplements the missing information in the original query. In AudioCaps, this method degrades the performance even if we add five queries. This result implies that the text queries of AudioCaps already include enough keywords for the retrieval task.

Ensemble model performance. We measure the performance of the ensemble of A, B, C, D, and E that have different audio encoders, which is similar to our system of the DCASE 2024 Challenge. Although our model is comparable with the best system of the DCASE 2024 Challenge for ClothoV2, do not outperform it for the evaluation data of the DCASE Challenge. This result shows that we cannot avoid overfitting by merely using the large-scale dataset and encoders.

6. CONCLUSION

This report shows the impact of the audio and text encoders, datasets, and data augmentation methods. In our experiments, the single model achieved 39.79 points and the ensemble of the models achieved 42.22 points in mAP@10 on average for the ClothoV2 benchmark. Our future work includes the training strategy for large-scale datasets and pre-trained that can avoid overfitting.

7. REFERENCES

[1] K. Drossos, S. Lipping, and T. Virtanen, “Clotho: An audio captioning dataset,” in *Proc. ICASSP*, 2020, pp. 736–740.

[2] A. van den Oord, Y. Li, and O. Vinyals, “Representation learning with contrastive predictive coding,” *arXiv preprint arXiv:1807.03748*, 2018.

[3] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark, G. Krueger, and I. Sutskever, “Learning transferable visual models from natural language supervision,” in *Proc. ICML*, 2021.

[4] C. Sun, A. Myers, C. Vondrick, K. Murphy, and C. Schmid, “VideoBERT: A joint model for video and language representation learning,” in *Proc. ICCV*, 2019.

[5] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of deep bidirectional transformers for language understanding,” in *Proc. ACL*, 2019, pp. 4171–4186.

[6] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, “RoBERTa: A robustly optimized bert pretraining approach,” *arXiv preprint arXiv:1907.11692*, 2019.

[7] S. Kim, G. Lee, S. Bae, and S.-Y. Yun, “Mixco: Mix-up contrastive learning for visual representation,” *arXiv preprint arXiv:2010.06300*, 2020.

[8] B. Elizalde, S. Deshmukh, and H. Wang, “Natural language supervision for general-purpose audio representations,” *arXiv preprint arXiv:2309.05767*, 2023.

[9] Y. Wu, K. Chen, T. Zhang, Y. Hui, T. Berg-Kirkpatrick, and S. Dubnov, “Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation,” in *Proc. ICASSP*. IEEE, 2023, pp. 1–5.

[10] K. Koutini, J. Schlüter, H. Eghbal-zadeh, and G. Widmer, “Efficient training of audio transformers with patchout,” in *Proc. INTERSPEECH*, 2022.

[11] S. Chen, Y. Wu, C. Wang, S. Liu, D. Tompkins, Z. Chen, W. Che, X. Yu, and F. Wei, “BEATs: Audio pre-training with acoustic tokenizers,” in *Proc. ICML*, 2023.

[12] Y. Gong, A. Rouditchenko, A. H. Liu, D. Harwath, L. Karlinsky, H. Kuehne, and J. R. Glass, “Contrastive audio-visual masked autoencoder,” in *Proc. ICLR*, 2022.

[13] S. Chen, H. Li, Q. Wang, Z. Zhao, M. Sun, X. Zhu, and J. Liu, “VAST: A vision-audio-subtitle-text omni-modality foundation model and dataset,” in *Proc. NeurIPS*, 2024.

[14] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, “Exploring the limits of transfer learning with a unified text-to-text transformer,” *J. Mach. Learn. Res.*, vol. 21, no. 140, pp. 1–67, 2020.

[15] C. D. Kim, B. Kim, H. Lee, and G. Kim, “AudioCaps: Generating captions for audios in the wild,” in *Proc. NAACL*, 2019.

[16] I. Martín-Morató, A. Mesaros, T. Heittola, T. Virtanen, M. Cobos, and F. J. Ferri, “Sound event envelope estimation in polyphonic mixtures,” in *Proc. ICASSP*. IEEE, 2019, pp. 935–939.

[17] 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.

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

[19] Y. Gong, Y.-A. Chung, and J. Glass, “AST: Audio spectrogram transformer,” in *Proc. INTERSPEECH*, 2021.

[20] H. Touvron, M. Cord, M. Douze, F. Massa, A. Sablayrolles, and H. Jégou, “Training data-efficient image transformers & distillation through attention,” in *Proc. ICML*, 2021, pp. 10 347–10 357.

[21] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly, *et al.*, “An image is worth 16x16 words: Transformers for image recognition at scale,” in *Proc. ICLR*, 2021.

[22] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, “Attention is all you need,” in *Proc. NeurIPS*, 2017.

[23] 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 *Proc. ICCV*, 2015.

[24] 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.

[25] H. Chen, W. Xie, A. Vedaldi, and A. Zisserman, “VGGSound: A large-scale audio-visual dataset,” in *Proc. ICASSP*, 2020, pp. 721–725.

[26] I. Loshchilov and F. Hutter, “Decoupled weight decay regularization,” *arXiv preprint arXiv:1711.05101*, 2017.