INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

Audio Retrieval with WavText5K and CLAP Training

Soham Deshmukh, Benjamin Elizalde, Huaming Wang

Microsoft

{sdeshmukh, benjaminm, huawang}@microsoft.com

Abstract

Text-based audio retrieval takes a natural language query to retrieve relevant audio files in a database. Most retrieval models are trained, optimized, and evaluated on a single dataset. In this paper, we quantify the effect of adding training data using three datasets and the effect on performance by evaluating the same model on two datasets. For our study, first, we introduce a new collection of about 5000 audio-text pairs called WavText5K. We qualitatively show how WavText5K differs from audio-text datasets and quantitatively show its effectiveness for retrieval. Our results show that adding more audio-text pairs does not necessarily improve performance. Second, we compare two effective audio encoders: CNN and audio transformers. We propose an architecture that demonstrates that utilizing both encoders improves the individual model's performance. Overall, using WavText5K and the proposed encoder combination outperforms the benchmark for AudioCaps and Clotho by 6% and 23%.

1. Introduction

The audio-retrieval technology has numerous applications in search engines, anomaly detection, and audio and video editing. We will use audio-retrieval to refer to both text-audio and audio-text retrieval in this work. The early works in audio-retrieval focused on using audio tags or events [5, 6, 7, 8]. With the availability of audio captioning datasets [1, 2, 4], the audio-retrieval task was expanded to include natural language descriptions as queries [9, 10]. The difficulty of building a high recall audio-retrieval system is evident from the metrics in the first audio retrieval Challenge at IEEE DCASE 2022 Task 6B[10].

The machine learning approach to audio-retrieval [4, 11, 9] consists of an audio encoder and a text encoder that learns a joint multimodal space. In literature, best results [11] are obtained by training independent audio-retrieval models for each target dataset. The direction of training on larger audio-text pairs and its impact on audio-retrieval performance is not explored. Therefore, in this work, we evaluate and quantify the effect of using multiple datasets in training without optimizing for a target evaluation dataset. We show that adding more training datasets does not necessarily improve audio-retrieval performance. We hypothesize that learning alignment between acoustic information and descriptions is difficult from complex audio scenes containing the occurrence of multiple audio events and interactions, leading to a drop in performance. So we introduce a new collection of audio-text pairs called WavText5K consisting of audio recordings and descriptions focused on isolated audio events. We describe what is isolated audio events and how these are different from the existing datasets in Section 2.2. From model architecture, the type of audio-encoder has a significant impact on audio-retrieval performance. We explore and compare two main families of audio encoders: CNN [12, 13] and Transformers [14, 15]. Recent literature has shown audio transformers [13, 16, 15, 14] work well on a variety of downstream tasks. However, the transformer models can intake limited input patches and tokens. This is unfavorable for training audio-retrieval models which have to learn temporal dependencies over 20-30 seconds of audio clips.

Our three main contributions are:

- We introduce a new 5000 audio-text pairs collection called WavText5K focused on isolated audio events and their descriptions.
- We quantify and show that adding more training data does not necessarily improve audio retrieval performance.
- Our proposed architecture combines two well-established audio encoders –a CNN and an audio transformer– outperforming benchmark performance. Hence, answering two questions unknown a priori. 1) Would combining a CNN and a transformer be redundant, or complement each other? 2) If they complement each other, what is the performance gain?

2. WavText5K

In this section, we introduce WavText5K and describe the unique sources of audio-text pairs, how the descriptions of the audio content differ from other datasets, and a quantitative analysis of the data.

2.1. Dataset collection

WavText5K is available online[1] and was sourced from two websites that have not been used in any other collection to the best of our knowledge: BigSoundBank[2] and SoundBible[3]. In Table 2, we compare WavText5K against the common datasets used for audio retrieval, particularly in the first audio retrieval Challenge at IEEE DCASE 2022 [10]. AudioCaps[1] is sourced from YouTube videos, Clotho [2] is sourced from freesound.org, MACS[3] is sourced from audio recorded in European locations, and SoundDescs[4] is sourced from the BBC website.

WavText5K is derived from two sound effects libraries that are royalty-free and free to download. The BigSoundBank website consists of sound effects in WAV, BFW, AIFF, MP3, OGG format with audio title and audio descriptions available. The SoundBible website consists of sound effects in WAV or MP3 with audio titles, descriptions. BigSoundBank has other metadata available like channels, conditions, sound type, bit depth etc, which we did not collect. The sampling rate of audio files varies, so we resampled all audio to 44.1 kHz. While collecting the audio, we encountered empty audio files, incorrect down-

[1] https://github.com/microsoft/WavText5K

[2] https://bigsoundbank.com [3] https://soundbible.com

10.21437/Interspeech.2023-1136

Dataset	Source	Language	Dur. (h)	Audios	Captions	Max dur.(s)	Avg dur.(s)	Max words	Avg words	Captions
AudioCaps [1]	YouTube	English	135.01	50535	55512	10.08	9.84	52	8.80	Human annotated
Clotho [2]	FreeSound	English	37.05	5929	29645	30.00	22.50	21	11.34	Human annotated
MACS [3]	TUT	English	11.88	3930	17275	10.88	10.88	40	9.24	Human annotated
SoundDescs [4]	BBC	English	1060.40	32979	32979	4475.38	115.75	65	15.28	Human-Automatic
WavText5K	BigSoundBank SoundBible	English	25.48	4525	4348	2438.65	20.27	82	12.50	User-generated

Table 1: *The captions of the first three datasets come from curated processes, SoundDescs' are obtained automatically from descriptions provided with the data, and WavText5K uses free-form descriptions provided by the uploader of the audio recording.*

load links, and empty metadata. We removed those entries from the final collection.

2.2. Description of isolated events

Source	Description
AudioCaps	*Screeching and light banging with a distant crow calling.*
Clotho	*A crow crows loudly as a person is heard imitating the sound.*
MACS	*a crow is screaming in the background kids are yelling then another bird is screaming.*
SoundDescs	*Birds - Birds, Madumbalai National Park, early morning with close-up partridge calls, warblers, crow-pheasants and house crow*
WavText5K	*A single crow crying in the middle of the night*

Table 2: *Variability of captions.*

WavText5K has audio-text pairs with natural language descriptions that focus on isolated events rather than complex acoustic content. In Table 2, we exemplify the variability in the complexity of captions across the datasets in this study. The captions in Clotho contain multiple sound events like "crow crows and a person imitating the crow" and the same in MACS "a crow screaming with kids yelling in the background along with another bird". SoundDescs has larger captions with multiple events in longer-duration audio recordings. In contrast to this, the WavText5K caption focuses on "crow crying" and provides a description of where and when ("the middle of the night") the event is happening. We show such audio-text pairs help in improving audio-retrieval performance in Section 5.2.

WavText5K has free-form descriptions provided by the uploader of the audio recording. The captions of Clotho or AudioCaps were designed primarily for audio captioning with curated annotation processes. SoundDescs was designed for audio retrieval benchmarks and is considerably more varied in duration and audio content than the previous two. However, the text descriptions are of mixed quality, since they are obtained automatically from descriptions provided with the original audio. MACS had a curated process to annotate descriptions but was not designed for any of these tasks. Due to the lack of captioned datasets MACS has become popular.

2.3. Data analysis

We provide a breakdown of the statistics of the datasets of this study in Table 1. WavText5K consists of 4525 audios, 4348 descriptions, and 4525 audio titles. The titles can be used together with the descriptions to form captions (see Section 5.4). The number of audios in WavText5K is comparable to MACS and Clotho. AudioCaps has the largest number of audios with 55,512 and SounDescs has 32,979. The average number of words in WavText5K descriptions (12.5 words) is comparable to other datasets. The average audio duration (20.27 seconds) is longer than datasets like AudioCaps and comparable to Clotho.

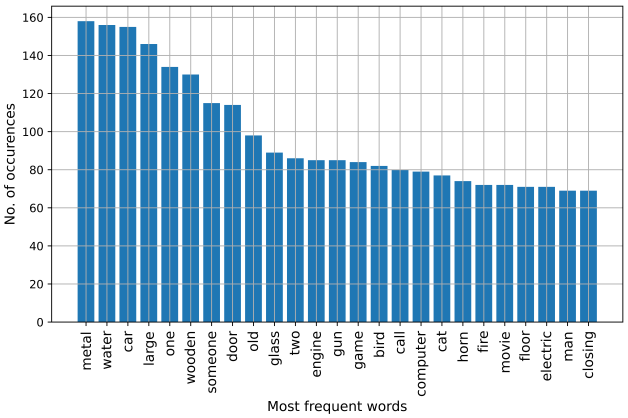

Figure 1: *Most frequent words in WavText5K descriptions.*

To exemplify some of the audio events encountered in WavText5K, we plot the most common words in Figure 1. We obtained the words from the descriptions, which are filtered to remove filler words. We can see that materials like "metal" and "water" are the two most common terms.

3. Audio-retrieval with contrastive learning

We utilize the CLAP model [17] which jointly trains audio and text encoder to learn a common multimodal space using contrastive learning. The trained audio and text encoder are then later used to retrieve files for audio-text and text-audio retrieval. The proposed architecture is shown in Figure 2.

3.1. CLAP Training

Let the training data be $D = \{(a_i, t_i)\}_{i=1}^{i=N}$. Let $f(a)$ be the audio encoder and $g(t)$ be the text encoder which are learnable embedding functions. Here, the audio encoder $f(a)$ first converts the raw audio into a log Mel spectrogram followed by a learnable embedding function. For a batch size of b:

$$x_a = \{f(a_i)\}_{i=1}^{i=b}; x_t = \{f(a_i)\}_{i=1}^{i=b} \quad (1)$$

where $x_a \in \mathbb{R}^{b \times v}$ are the audio representations of dimension v, and $x_t \in \mathbb{R}^{b \times u}$ are the text representations of dimension u. The audio and text representation are brought into a common multimodal space of dimension d by independent linear projection layers $l_a(a)$, $l_t(t)$. This results in:

$$\hat{x}_a = l_a(x_a); \hat{x}_t = l_t(x_t) \quad (2)$$

where $\hat{x}_a \in \mathbb{R}^{b \times d}$ and $\hat{x}_t \in \mathbb{R}^{b \times d}$. Once both audio and text embeddings are in common embedding space, we can compare their similarity as:

$$C = \tau(\hat{x}_a \cdot \hat{x}_t^\top) \quad (3)$$

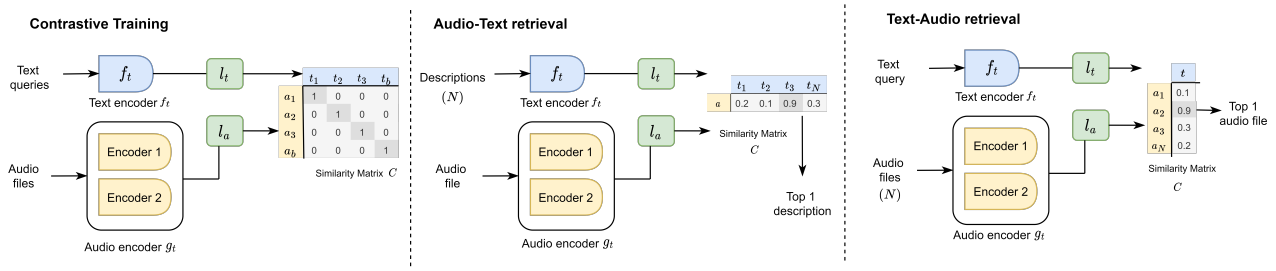

Figure 2: *The model is trained on audio-text pairs using CLAP. At testing/retrieval time, the trained encoders match the audio query to descriptions (audio-text retrieval) or text query to audio files (text-audio retrieval) in the database.*

where τ is a temperature parameter and the similarity matrix C has b correct pairs in the diagonal. To learn the embedding functions and projection layers we use symmetric cross-entropy loss ($\mathcal{L}$):

$$\mathcal{L} = 0.5(\ell_{text}(C) + \ell_{audio}(C)) \tag{4}$$

where $\ell_k = \frac{1}{N}\sum_{i=0}^{N} \log diag(softmax(C))$ along text and audio axis respectively

3.2. Audio Retrieval

After CLAP training, the model is used for audio-retrieval as shown in Figure 2. The audio file(s) are embedded by audio encoder f_t and the descriptions by text encoder g_t. This is followed by independently projecting (l_t, l_a) the embeddings into common multimodal space and computing the similarity matrix between the audio and text embeddings. This similarity matrix is represented as C in Figure 2. For audio-text retrieval, top-N descriptions are computed by picking the descriptions corresponding to the top N values in similarity matrix C. Similarly, for text-audio retrieval, top-N audios are computed by picking the audios corresponding to the top N values in C.

3.3. Audio and Text encoder

The CLAP model [17] used PANN's CNN14[13] as the audio encoder and BERT as the text encoder. There have been recent advances in audio transformer models [16, 15, 14] which show comparable or better performance than CNN models. However, the transformer models can intake limited input patches and tokens. For example, the HTSAT is trained with 10 seconds audio clips. This is unfavorable for audio-text pair training where the audio concepts and complex descriptions have temporal dependencies which evolve over 20-30 seconds of audio. On the other hand, the HTSAT (0.47 mAP) is better than CNN14 (0.38 mAP) in understanding sound events. So instead, we propose using the combination of CNN14 and HTSAT as the encoder:

$$f(a) = \text{Concat}(CNN14(a), HTSAT(a)) \tag{5}$$

In section 5.3, we show that the CNN14 and HTSAT complement each other and improve audio-retrieval performance. We leave the investigation of unified models like Wav2Vec [18, 19] and deep fusion for future work. For text encoder, we use RoBERTa [20] instead of BERT which is a more robust text encoder. We leave dynamic methods of combining audio embeddings like attention mechanisms [21, 22] for future work.

4. Experiments

4.1. Datasets

We use Clotho [2] and AudioCaps [1] for training the baseline and their test sets for audio-retrieval model evaluation. We also use MACS [3], SoundDescs [4] and WavText5K in the training dataset for experiments in Table 3. For WavText5K, we use 4348 audio-text pairs which have both description and title.

4.2. Experimental setups

The audio files are resampled to 44.1 kHz and represented by log Mel spectrogram. The log Mel spectrogram is constructed with a hop size of 320, a window size of 1024, and 64 Mel bins in the range of 50-8000 Hz. Each audio file is randomly truncated to 20 secs for CNN14 and 10 secs for HTSAT. We use SpecAugment [23] for augmenting audio files. The captions were not augmented or altered. The audio-text pairs are randomly sampled to form batches during training. The projection dimension for CLAP is set to be 1024 and the temperate τ is initialised to 0.007. We use Adam Optimiser [24] with the learning rate of 10^{-4} which is reduced by a factor of 0.1 every 20 epochs for a total of 45 epochs. The model is trained on 8 GPUs with a batch size of 128.

5. Results and Discussion

In this section, we show how WavText5K helps to improve audio-retrieval performance. We evaluate and quantify the effect of training an audio retrieval system on multiple datasets. We show how our proposed architecture combining encoders outperforms the benchmark.

5.1. Training with multiple datasets

Our baseline is a model trained on AudioCaps and Clotho and the test set of either AudioCaps or Clotho are used to evaluate the model's performance for audio to text and text to audio retrieval. As noted in [4], the Clotho dataset is particularly more challenging than AudioCaps due to its varied audio content distributed in 10-30 seconds audio files. This is unlike AudioCaps which is limited in temporal dependencies to 10 seconds.

In Table 3 we show that adding more data (audio-text pairs) to training does not necessarily improve performance. We used CNN14 as an audio encoder and kept settings constant across experiments. Adding MACS [3] (row 2 and row 6 in Table 3), adds 17k audio-text pairs and leads to a drop in both audio-text and text-audio retrieval performance. SoundDescs has 33k recordings consisting of complex acoustic scenes and detailed descriptions. The model is not able to utilize and learn from SoundDescs audio-text pairs, as evident from row 4 and row 8 in Table 3. Authors in [25, 26, 27] reported similar conclusions with self-collected audio-text pairs and did not provide numerical evidence to understand the effect in performance.

5.2. WavText5K improves performance

We hypothesize that learning alignment between acoustic information and descriptions is difficult from complex audio scenes

Training dataset	Training pairs (k)	Retr. dataset	Text-Audio Retrieval ↑					Audio-Text Retrieval ↑				
			mAP@10	R@1	R@5	R@10	R@50	mAP@10	R@1	R@5	R@10	R@50
AC, Cl	65k	AC	45.28	33.07	67.30	80.30	95.74	25.65	39.76	73.72	84.64	97.04
AC, Cl, MACS	82k	AC	44.42	30.91	65.10	79.10	94.71	24.88	37.33	68.73	81.67	96.22
AC, Cl, SD	98k	AC	45.03	31.37	66.60	79.32	95.19	25.21	33.82	68.46	82.21	96.90
AC, Cl, WT5K	70k	AC	**46.57**	**33.42**	**68.00**	79.95	**96.42**	**26.30**	38.68	70.35	84.1	**97.44**
AC, Cl	65k	Cl	24.74	15.79	36.78	49.93	80.75	12.41	17.42	40.57	54.26	82.68
AC, Cl, MACS	82k	Cl	24.55	14.79	37.60	48.52	80.10	11.61	16.45	38.94	52.72	81.92
AC, Cl, SD	98k	Cl	23.85	14.39	35.73	49.14	80.03	11.88	17.71	40.19	54.02	84.01
AC, Cl, WT5K	70k	Cl	**25.85**	**16.48**	**39.58**	**52.46**	**82.00**	**12.87**	**18.47**	**44.02**	**57.51**	**86.03**

Table 3: *Experiments of training with different datasets: AC (AudioCaps), Cl (Clotho), MACS, SD (SoundDescs), WT5K (WavText5K). All experiments use the same setting for training CLAP model with CNN14 audio encoder. R is Recall and mAP is mean Average Precision*

Exp - Audio encoder	Training dataset	Retr. dataset	Text-Audio Retrieval ↑					Audio-Text Retrieval ↑				
			mAP@10	R@1	R@5	R@10	R@50	mAP@10	R@1	R@5	R@10	R@50
Benchmark [11]	AC	AC	-	33.90	69.70	82.60	-	-	39.40	72.0	83.90	-
CNN14	AC, Cl, WT5K	AC	46.57	33.42	68.00	79.95	96.42	26.30	38.68	70.35	84.10	97.44
HTSAT	AC, Cl, WT5K	AC	46.33	34.07	66.90	79.81	95.36	26.71	40.84	72.77	84.36	97.30
CNN14-HTSAT	AC, Cl, WT5K	AC	**49.45**	**34.69**	**70.22**	**82.00**	**97.28**	**30.81**	**41.91**	**73.18**	**84.64**	**97.71**
Benchmark [11]	Cl	Cl	-	14.40	36.60	49.90	-	-	16.20	37.50	50.20	-
CNN14	AC, Cl, WT5K	Cl	25.85	16.48	39.58	52.46	82.00	12.87	18.47	44.02	57.51	86.03
HTSAT	AC, Cl, WT5K	Cl	22.62	14.24	36.11	49.29	82.47	10.15	16.36	38.37	50.43	81.15
CNN14+HTSAT	AC, Cl, WT5K	Cl	**27.12**	**16.75**	**41.09**	**54.07**	**83.79**	**13.65**	**20.0**	**44.88**	**58.66**	**87.65**

Table 4: *The CNN and Transformer audio encoder complement each other and outperform the literature benchmark [11].*

containing the occurrence of multiple audio events and interactions, leading to a drop in performance. So we evaluated WavText5K as training data in row 4 and row 8 in Table 3. With about 5000 audio-text pairs focusing on isolated events, both text-audio and audio-text performance improves. For the harder Clotho evaluation, text-audio R@1 improves by 4.4% and audio-text retrieval R@1 improves by 6%.

5.3. Audio encoder architecture

In Table 4, we show that CNN and the audio transformer complement each other and improve audio retrieval performance. CNN is the most common encoder in the audio retrieval literature, followed by audio transformers [28, 27]. However, they have not been combined into one architecture. Other authors have combined up to four encoders [25], but have not outperformed benchmarks like our architecture. In row 3 of Table 4, the audio retrieval model with transformer-based audio encoder HTSAT performs better than the CNN14 model on AudioCaps evaluation. However, the CNN-based audio encoder (row 6) performs better on Clotho evaluation where the recording is 20-30 seconds in length and more complex than AudioCaps. By combining CNN14 and HTSAT, the audio retrieval models' performance increases on all metrics for both the evaluation datasets. Compared with benchmark [11] on row 1, our proposed combination leads to an improvement on text-audio R@1 by 2.3% and 16.3% on AudioCaps and Clotho dataset respectively. Similarly, for audio-text R@1, we see an improvement of 6.4% and 23.5% on AudioCaps and Clotho respectively.

5.4. Caption construction in WavText5K

Titles in WavText5K have additional information which can be combined with the descriptions to improve audio-retrieval performance (See Table 5). Overall we suggest using the title and descriptions for training models. We performed an ablation study to understand the effect of caption construction using *"{title}"* and *"{description}"* in addition to AudioCaps and Clotho to train the model CNN14+HTSAT. On Clotho, using

	Retrieval dataset	T-A Retrieval ↑		A-T Retrieval ↑	
		mAP@10	R@1	mAP@10	R@1
Desc.	AC	**49.54**	**34.77**	30.48	40.16
Desc., Title	AC	49.45	34.69	**30.81**	**41.91**
Desc.	Cl	26.15	16.19	12.77	19.14
Desc., Title	Cl	**27.12**	**16.75**	**13.65**	**20.0**

Table 5: *Constructing the caption with the title and the description of WavText5K results in better retrieval performance.*

caption as *"{title}. {description}"* provides about 3.5% and 4.5% improvement on T-A and A-T R@1 metrics against using only *"{description}"* as caption. On AudioCaps, we observed mixed results where performance decreases by 0.2% on T-A retrieval and increases on A-T retrieval by 4.4%.

6. Conclusion

We analyzed the effect of training audio-retrieval system on multiple datasets and quantified its effect on audio-retrieval performance on two publicly available datasets. We found that adding more audio-text pairs or training datasets does not necessarily improve audio-retrieval performance. We introduced a collection of 5000 audio-text pairs called WavText5K which focus on isolated audio events and their descriptions. We demonstrate quantitatively how adding WavText5K to training data improves audio-text and text-audio retrieval performance, unlike other datasets like SoundDescs and MACS. Our analysis of audio encoder architecture shows that CNN and Transformer models complement each other, and their combination achieves benchmark performance on both audio-retrieval evaluation datasets. Further exploration of the relation between the *quality* of audio-text pairs and its effect on learning audio representations can offer additional insights into making text-based retrieval systems better.

7. References

[1] C. D. Kim, B. Kim, H. Lee, and G. Kim, "AudioCaps: Generating Captions for Audios in The Wild," in *NAACL-HLT*, 2019.

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

[3] I. Martín-Morató and A. Mesaros, "What is the ground truth? reliability of multi-annotator data for audio tagging," in *2021 29th European Signal Processing Conference (EUSIPCO)*. IEEE, 2021, pp. 76–80.

[4] A. S. Koepke, A.-M. Oncescu, J. Henriques, Z. Akata, and S. Albanie, "Audio retrieval with natural language queries: A benchmark study," *IEEE Transactions on Multimedia*, 2022.

[5] G. Chechik, E. Ie, M. Rehn, S. Bengio, and D. Lyon, "Large-scale content-based audio retrieval from text queries," in *Proceedings of the 1st ACM International Conference on Multimedia Information Retrieval*, ser. MIR '08. New York, NY, USA: Association for Computing Machinery, 2008, p. 105–112. [Online]. Available: https://doi.org/10.1145/1460096.1460115

[6] S. Ikawa and K. Kashino, "Acoustic event search with an onomatopoeic query: measuring distance between onomatopoeic words and sounds," in *DCASE*, 2018.

[7] B. Elizalde, S. Zarar, and B. Raj, "Cross modal audio search and retrieval with joint embeddings based on text and audio," in *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2019, pp. 4095–4099.

[8] B. M. Elizalde, "Never-ending learning of sounds," Ph.D. dissertation, Carnegie Mellon University Pittsburgh, PA, 2020.

[9] A.-M. Oncescu, A. S. Koepke, J. F. Henriques, Z. Akata, and S. Albanie, "Audio Retrieval with Natural Language Queries," in *Proc. Interspeech 2021*, 2021, pp. 2411–2415.

[10] H. Xie, S. Lipping, and T. Virtanen, "Dcase 2022 challenge task 6b: Language-based audio retrieval," *arXiv preprint arXiv:2206.06108*, 2022.

[11] X. Mei, X. Liu, J. Sun, M. D. Plumbley, and W. Wang, "On metric learning for audio-text cross-modal retrieval," *arXiv preprint arXiv:2203.15537*, 2022.

[12] S. Hershey, S. Chaudhuri, D. P. W. Ellis, J. F. Gemmeke, A. Jansen, C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold, M. Slaney, R. Weiss, and K. Wilson, "Cnn architectures for large-scale audio classification," in *International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2017. [Online]. Available: https://arxiv.org/abs/1609.09430

[13] Q. Kong, Y. Cao, T. Iqbal, Y. Wang, W. Wang, and M. D. Plumbley, "Panns: Large-scale pretrained audio neural networks for audio pattern recognition," *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, vol. 28, p. 2880–2894, jan 2020. [Online]. Available: https://doi.org/10.1109/TASLP.2020.3030497

[14] K. Chen, X. Du, B. Zhu, Z. Ma, T. Berg-Kirkpatrick, and S. Dubnov, "Hts-at: A hierarchical token-semantic audio transformer for sound classification and detection," in *ICASSP 2022 - 2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2022, pp. 646–650.

[15] Y. Gong, C.-I. Lai, Y.-A. Chung, and J. Glass, "Ssast: Self-supervised audio spectrogram transformer," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 10, 2022, pp. 10 699–10 709.

[16] Y. Gong, Y.-A. Chung, and J. Glass, "AST: Audio Spectrogram Transformer," in *Proc. Interspeech 2021*, 2021, pp. 571–575.

[17] B. Elizalde, S. Deshmukh, M. A. Ismail, and H. Wang, "Clap: Learning audio concepts from natural language supervision," *arXiv preprint arXiv:2206.04769*, 2022.

[18] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *Advances in Neural Information Processing Systems*, vol. 33, pp. 12 449–12 460, 2020.

[19] S. Schneider, A. Baevski, R. Collobert, and M. Auli, "wav2vec: Unsupervised Pre-Training for Speech Recognition," in *Proc. Interspeech 2019*, 2019, pp. 3465–3469. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2019-1873

[20] 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.

[21] S. Deshmukh, B. Raj, and R. Singh, "Improving Weakly Supervised Sound Event Detection with Self-Supervised Auxiliary Tasks," in *Proc. Interspeech 2021*, 2021, pp. 596–600.

[22] ——, "Multi-task learning for interpretable weakly labelled sound event detection," *arXiv preprint arXiv:2008.07085*, 2020.

[23] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "SpecAugment: A Simple Data Augmentation Method for Automatic Speech Recognition," in *Proc. Interspeech 2019*, 2019, pp. 2613–2617. [Online]. Available: http://dx.doi.org/10.21437/Interspeech.2019-2680

[24] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," in *ICLR (Poster)*, 2015. [Online]. Available: http://arxiv.org/abs/1412.6980

[25] T. L. de Gail and D. Kicinski, "Take it easy: Relaxing contrastive ranking loss with CIDEr," DCASE2022 Challenge, Tech. Rep., July 2022.

[26] . Benno Weck1, . Miguel Pérez Fernández1, H. Kirchhoff1, and X. Serra2, "Aligning audio and text embeddings for the language-based audio retrieval task of the DCASE challenge 2022," DCASE2022 Challenge, Tech. Rep., July 2022.

[27] X. Xu, Z. Xie, M. Wu, and K. Yu, "The SJTU system for DCASE2022 challenge task 6: Audio captioning with audio-text retrieval pre-training," DCASE2022 Challenge, Tech. Rep., July 2022.

[28] Yusong Wu1, 2, Tianyu Zhang1, 2, and Ke Chen3, "Text-to-audio retrieval via large-scale contrastive training," DCASE2022 Challenge, Tech. Rep., July 2022.