Leveraging Pre-trained AudioLDM for Sound Generation: A Benchmark Study

Yi Yuan[1*], Haohe Liu[1*], Jinhua Liang[2], Xubo Liu[1], Mark D. Plumbley[1], Wenwu Wang[1],

[1]Centre for Vision, Speech and Signal Processing (CVSSP), University of Surrey, UK
[2]Centre for Digital Music (C4DM), Queen Mary University of London

***Abstract*—Deep neural networks have recently achieved breakthroughs in sound generation. Despite the outstanding sample quality, current sound generation models face issues on small-scale datasets (e.g., overfitting), significantly limiting performance. In this paper, we make the first attempt to investigate the benefits of pre-training on sound generation with AudioLDM, the cutting-edge model for audio generation, as the backbone. Our study demonstrates the advantages of the pre-trained AudioLDM, especially in data-scarcity scenarios. In addition, the baselines and evaluation protocol for sound generation systems are not consistent enough to compare different studies directly. Aiming to facilitate further study on sound generation tasks, we benchmark the sound generation task on various frequently-used datasets. We hope our results on transfer learning and benchmarks can provide references for further research on conditional sound generation.**

***Index Terms*—Sound generation, Auditory evaluation, Benchmark system, Pre-trained networks, Transferring network**

I. Introduction

The development of deep learning models has led to a surge of interest in sound generation. Different strategies have been developed for sound generation tasks with input contents as diverse as tag [1], text [2], [3] and video [4]. Sound generation systems are useful tools for content creation in applications such as virtual reality, movies, music, and digital media [5]–[7].

Recently, significant progress has been made in high-fidelity text-to-sound generation [2], [8], [9]. Such sound generation systems are usually data-hungry to train. For example, AudioGen [2] collected ten different datasets for training. However, this is not viable in some real-world applications, (e.g., animal sound and environmental sound generation), where the collection and labelling work for this specific domain is a time-consuming and costly process, leading to limited datasets scale in practice. How to overcome the data scarcity issue is a significant challenge in sound generation research. Several methods are adapted to tackle this issue. Rongjie *et al.* [10] introduced a pseudo prompt enhancement approach to increase the data quantity, while the quality of new augmented data is unstable. Given these considerations, it is intuitive to ask: *can we find an effective solution to train a sound generative model with a small-scale dataset?*

Studies have shown that pre-trained models can achieve faster and more accurate adaptions in tasks with limited data [11], [12]. Concretely, pre-trained models are neural networks that have already trained on a massive corpus and can be fine-tuned into downstream tasks. Over the last few years, pre-training strategies have achieved enormous success across multiple fields including text [13]–[15], image [16], [17] and audio [18]–[20]. However, the effectiveness of a pre-trained model for sound generation is an under-explored topic. This paper takes the first step on investigating the effectiveness and feasibility of pre-training in text-to-sound generation with AudioLDM [9], the state-of-the-art audio generation model. Our results show that pre-trained models can achieve better performance on sound generation, especially for small-scale datasets.

Besides, previous sound generation studies used different methodologies for evaluation, making it difficult for us to evaluate the model performance fairly. Without a set of constant metrics, researchers may find it hard to reproduce the results of other models. Aiming to provide an efficient and reliable reference for further sound generation research, this paper introduces a new benchmark with pre-trained AudioLDM on four commonly used audio datasets: AudioCaps [21], AudioSet [22], Urbansound8K (US8K) [23] and ESC50 [24]. Furthermore, our new benchmark contains most of the evaluation metrics applied in previous works [2], [8], [9], including Fréchet Distance (FD), Inception Score (IS) [25], Fréchet Audio Distance (FAD) [26] and Kullback-Leibler (KL) divergence. With several qualitative experiments, we provide insights into the effectiveness of these metrics in evaluating sounds. Our contributions are as follows:

- We showcase that transferring the pre-trained AudioLDM is beneficial for sound-generation tasks in both sample quality and training efficiency, especially for small-scale datasets.
- We benchmark the sound generation task by presenting the result of AudioLDM with multiple settings on four commonly used sound datasets.

II. Related Work

Conditional sound generation. Kong *et al.* [27] took the first step on conditional generation by taking labels as input and generating waveforms with recurrent neural network (RNN). Then, Liu *et al.* [1] tried to synthesise sound with latent discrete features obtained from a vector quantised-variational autoencoder (VQ-VAE) [28] in the frequency domain (e.g. mel-spectrogram). By compressing the mel-spectrogram into a sequence of tokens, the model can generate sounds with long-range dependencies. Recently, remarkable progress has been made in text-to-sound generations. Diffsound [8] explores

* Equal contributions

generating audio with a diffusion-based text encoder, a VQ-VAE-based decoder and a generative adversarial network (GAN)-based vocoder. Taking texts as input, Diffsound utilized a contrastive language image pre-training (CLIP) model [29] for text embedding before sending it to the encoder. To alleviate the scarcity of text-audio pairs, they proposed a text-generating strategy by combining mask tokens and sound labels. AudioGen [2] used a similar encoder-decoder structure to Diffsound [8], while generating waveform directly instead of using a vocoder. They used a transformer-based encoder to generate discrete tokens and a pre-trained Transfer Text-to-Text Transformer (T5) [30] for text embedding. To increase the quantity of sound, they mixed audio samples at various signal-to-noise ratios (SNR) and collect 10 large datasets.

Evaluation metrics for sound generation. Since subjective metrics for sound-generating systems usually require a huge amount of time and workload, various objective metrics were applied for this task. However, previous works often adopted different evaluation metrics, which makes it difficult to compare them in a common ground. Kong *et al.* [27] used Inception Score [25] as the criterion. Liu *et al.* [1] trained a sound classifier to verify the sample quality. Diffsound [8] applied Fréchet Inception Distance (FID) [31] and Kullback-Leibler (KL) divergence to compute the sample fidelity, as well as a pre-trained audio caption transformer (ACT) to calculate a sound-caption-based loss. AudioGen [2] evaluated the result with KL divergence and Fréchet Audio Distance (FAD).

III. Method and Datasets

A. AudioLDM

Given text-based information (i.e. a written description containing single or multiple sound events), the objective of text-to-sound generation is to generate an audio clip that presents correct sound events as the text description.

Our experiments are carried out with AudioLDM [9], a continuous latent diffusion-based model (LDMs) for text-to-sound generations. Inspired by previous text-to-sound models, AudioLDM adapts a similar encoder, decoder, and vocoder architecture. By comparison, the text encoder in previous studies [2], [8], [10] is replaced by a Contrastive Language-Audio Pre-training (CLAP) model. Specifically, the CLAP consists of two encoders, a text encoder f_{text} that extracts text description y into text embedding $\boldsymbol{E}^y$ and an audio encoder f_{audio} that computes audio embedding $\boldsymbol{E}^x$ from audio samples x. CLAP trains two encoders along with two projection layers using a symmetric cross-entropy loss, resulting in an aligned audio-text latent space. By utilizing the audio embedding during training and text embedding during sampling, AudioLDM can significantly reduce the demand for text-sound pairs and enable a self-supervised paradigm of LDM optimization. The latent diffusion model contains two processes: 1) a forward process that gradually transforms the data into a standard Gaussian distribution; and 2) a reverse process that generates data from the Gaussian distribution by denoising in reverse order as the forward process. During the forward process, the continuous latent representation $\boldsymbol{z}_0$ from the mel-spectrogram is transformed into a standard Gaussian distribution $\boldsymbol{z}_n$ by gradually adding a scheduled Gaussian noise in N steps. The transition probability of each time step n is:

$$q(\boldsymbol{z}_n|\boldsymbol{z}_{n-1}) = \mathcal{N}(\boldsymbol{z}_n; \sqrt{1-\beta_n}\boldsymbol{z}_{n-1}, \beta_n \boldsymbol{I}), \quad (1)$$

$$q(\boldsymbol{z}_n|\boldsymbol{z}_0) = \mathcal{N}(\boldsymbol{z}_n; \sqrt{\bar{\alpha}_n}\boldsymbol{z}_0, (1-\bar{\alpha}_n)\boldsymbol{\epsilon}), \quad (2)$$

where $\epsilon \sim \mathcal{N}(0, \boldsymbol{I})$ denotes the Gaussian noise with level α_n and schedule β_n. The latent diffusion model is trained with the re-weighted training objective [9], [32], given by

$$L_n(\theta) = \mathbb{E}_{\boldsymbol{z}_0,\boldsymbol{\epsilon},n} \left\| \boldsymbol{\epsilon} - \boldsymbol{\epsilon}_\theta(\boldsymbol{z}_n, n, \boldsymbol{E}^x) \right\|_2^2, \quad (3)$$

where θ denotes the trainable parameters in the LDMs. Benefits from the aligned audio-text space from CLAP, the reverse transition probability, $p_\theta(\boldsymbol{z}_{n-1}|\boldsymbol{z}_n, \boldsymbol{E}^y)$, can be parameterized by both $\boldsymbol{\epsilon}_\theta(\boldsymbol{z}_n, n, \boldsymbol{E}^y)$ and $\boldsymbol{\epsilon}_\theta(\boldsymbol{z}_n, n, \boldsymbol{E}^x)$ [9]. Data can be generated by performing reverse diffusion from a sample of standard Gaussian distribution with the reverse transition probability [32]. We will compare the difference between conditioning with $\boldsymbol{E}^x$ and $\boldsymbol{E}^y$ in our experiment.

B. Dataset

Dataset for pre-training AudioLDM. The datasets for pertaining AudioLDM include AudioSet [22], AudioCaps [21], Freesound[1], and BBC Sound Effect library (BBC SFX)[2]. AudioSet [22] is the largest dataset with 527 text labels and around 5000 hours of sound. AudioCaps [21] is a smaller dataset with additional human-written audio captions. Both AudioCaps [21] and AudioSet [22] are captured from YouTube. FreeSound is a dataset provided by a public sound community with various durations. BBC SFX is a high-quality dataset from BBC with a wide range of sound effects. Note that most pre-training datasets originally come with text-sound pairs (e.g. AudioCaps and BBC SFX), we only use audio embedding as the condition during the self-supervised training of LDMs. Combining all four datasets, we have totally 3.3M ten-second sound clips to train our AudioLDM.

Dataset for benchmark study. To establish the baselines for the transferring study, we perform experiments on three common audio datasets with different volumes.

Two relatively small datasets we use are Urbansound8K (US8K) [23] and ESC50 [24]. US8K contains 8000 sound clips with 10 classes and ESC50 has 50 classes with only 40 samples for each class. We randomly select 870 samples in US8K and 400 samples in ESC50 for evaluation. Apart from ESC50 and Urbansound8K, we also perform experiments on AudioCaps [21] to further enhance our study. AudioCaps contains around 47000 ten-second audio data with more diverse sound events. Although AudioCaps is included in the pre-training dataset of AudioLDM, we find further fine-tuning on AudioCaps can improve model performance on the AudioCaps evaluation set.

[1] https://freesound.org/

[2] https://sound-effects.bbcrewind.co.uk/search

IV. Evaluations and Experiments

A. Evaluation

The evaluation is performed between a set of generated audio and a set of target audio files. For model evaluation, we follow the metrics used by AudioLDM, including Fréchet Distance (FD), Inception Score (IS), Fréchet Audio Distance (FAD), and Kullback–Leibler (KL) divergence. All four metrics are calculated based on logits or embedding from audio classifiers. Specifically, IS calculates the entropy of label distribution, where a higher IS indicates a larger variety with vast distinction. KL divergence measures the similarity between generated and target audio by comparing the logits distributions. FAD first computes the multivariate Gaussian of two embedding values collected from a pre-trained VGGish [33]. Then, this score calculates the Fréchet distance between the Gaussian mean and variance. Both KL and FAD indicate better fidelity with lower scores. Besides the three common practices (IS, FAD and KL) used in previous works [2], [8], [27], we also adopt FD, which has a similar idea with FAD but uses PANNs [34], a pre-trained audio pattern recognition model, as the backbone classifier for feature embedding. To compare the effectiveness of these metrics, we perform evaluations between a set of audio files and their corrupted version by the following:

(1) *Adding noise and random masking.* We add Gaussian noise and mask some information on the mel-spectrogram domain. For Gaussian noise, the mean is the mean value of the mel-spectrogram and the variance equals to 20% of the value range. For masking, we randomly select two places and mask the value of a 10% length of overall mel-spectrogram (e.g., setting a length of 86 into zero for a 860 long mel-spectrogram). As Figure 1 shows, all the metrics can detect this change with a rapid fall or rise.

(2) *Adding interference sound.* We randomly select ten irrelevant classes of audio clips and mix them directly with the target sound under the same SNR on mel-spectrogram to verify whether these interfered sounds can be detected. As shown in Figure 1, it is obvious that KL and IS do not present significant changes. This might be because adding interfering sounds does not lead to a distinct change in the sound quality. In comparison, FD and FAD can effectively detect changes with an apparent increase in scores.

(3) *Changing order.* We testify to the sensitivity of these metrics when acoustic events are placed in the wrong order. To simulate this change, the ground-truth data is composed of a group of different sound events and we randomly change their orders. Figure 1 shows that with the increase of disordered events, only FD presents an increasing trend while other metrics stay stable with little fluctuations. Based on the qualitative findings, all four metrics detect the noise efficiently, while only the FD score is capable on classifying irrelevant sounds.

B. Benchmark Study

As shown in Table I, we evaluate the performance of pre-trained AudioLDM[3] as our baselines for text-to-sound generations. Note that we do not perform any fine-tuning on AudioLDM in this section. Although the open-sourced version of AudioLDM is trained with audio embeddings, AudioLDM can perform sampling with either audio or text embedding. Table I shows the effect of conditions on different modalities, where AudioLDM conditioned with audio embedding performs better than text embedding in most cases. This indicates the distribution of audio and text embedding is not completely aligned, and audio embedding is a more precise conditioning signal for sound generation.

[3]https://github.com/haoheliu/AudioLDM

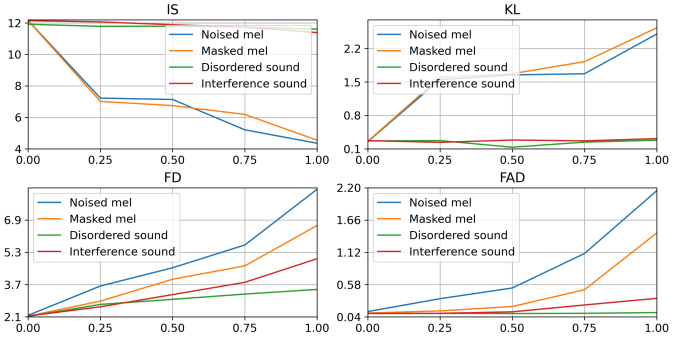

Fig. 1. The metrics are evaluated with the increase of the percentage (from 0 to 1) of pre-processed data on: 1) adding noise on the mel-spectrogram; 2) masking value on the mel-spectrogram; 3) making disorder sound events; 4) adding interfering sound events. Regarding metrics capabilities, higher IS and lower KL, FD, and FAD indicate better sample quality.

TABLE I
The baseline of four datasets on pre-trained AudioLDM

Dataset	Test Condition	FD↓	IS↑	KL↓	FAD↓
ESC50	Text	60.63	5.55	3.01	5.95
	Audio	47.46	6.68	2.08	4.81
US8K	Text	31.20	3.88	2.20	10.00
	Audio	32.79	4.04	1.44	13.74
AudioCaps	Text	23.63	6.68	2.36	4.94
	Audio	21.37	6.65	1.78	2.18
AudioSet	Text	20.30	7.56	2.34	4.26
	Audio	19.04	6.72	1.63	1.52

C. Fine-tuning Study

To study the effectiveness of the pre-trained audio generative model, we fine-tune and evaluate the pre-trained AudioLDM on three smaller datasets, US8K, ESC50 and AudioCaps. Author of [9] finds audio embedding is better than text embedding in some cases as model condition information. To validate this conclusion in more datasets, we adopt a similar experiment setting and fine-tune AudioLDM with both text embedding and audio embedding as conditioning information. Different from Table I, all the experiments in this section only sample with text embedding as input condition since we mainly focus on text-to-sound generation. During the fine-tuning process, we freeze the parameter of CLAP and the VAE encoder, leaving only the latent diffusion model for training. To validate the effect of model pre-training, we also train and evaluate AudioLDM on different datasets from scratch.

TABLE II

THE COMPARISON BETWEEN DIFFERENT PRE-TRAINED STRATEGIES. EXPERIMENTS WITHOUT PRE-TRAINING INVOLVE BUILDING NEW MODELS FROM SCRATCH. AUDIO AND TEXT INDICATE WHETHER THE MODEL IS TAKING AUDIO EMBEDDING OR TEXT EMBEDDING AS THE CONDITION IN TRAINING. ALL THE RESULTS SHOW THE BEST SCORE DURING TRAINING.

Dataset	Pre-training	Train Condition	Train Steps (K)	FD ↓	IS ↑	KL ↓	FAD ↓
ESC50	✗	Audio	240	44.75	7.44	3.31	4.02
	✗	Text	160	30.74	10.22	1.84	3.28
	✓	Audio	180	36.43	11.15	2.15	4.41
	✓	Text	80	**22.38**	**12.98**	**1.56**	**2.66**
US8K	✗	Audio	160	33.69	3.73	2.04	5.75
	✗	Text	350	28.45	**5.00**	**1.87**	**4.45**
	✓	Audio	20	31.21	3.84	2.11	7.39
	✓	Text	240	**28.44**	4.91	1.88	4.88
AudioCaps	✗	Audio	480	24.04	7.12	2.20	2.98
	✗	Text	480	24.84	6.91	2.25	2.47
	✓	Audio	80	**23.57**	7.21	**2.09**	2.98
	✓	Text	240	25.78	**7.95**	2.26	**1.67**

Table II shows the experiment results of this fine-tuning study. We notice that the pre-trained AudioLDM is more advantageous than the model trained from scratch in most cases. With only 32 samples in each class, the performance of ESC50 can be significantly improved with pre-training. On US8K, the performance of pre-trained AudioLDM is slightly lower, which might attribute to 1) US8K is large enough for model optimization, with around 800 samples for each class; 2) US8K only contains 10 sound classes while the pre-trained AudioLDM is capable of generating sound with more diversity, which might degrade model performance on US8K evaluation set. Additionally, the pre-trained model can improve generation quality on AudioCaps, particularly on the FAD scores. We also notice that fine-tuning with text embedding on AudioCaps can further achieve a better IS score. We analyze the reason as text embeddings provide weaker conditions, leading to results with less restriction and more diversity.

AudioLDM is trained in a self-supervised way using audio embedding as conditioning information because training data can be easily scaled up with this scheme. AudioLDM also found that taking audio embedding as the training condition is better than text embedding. However, our experiment shows this is not always the case on different datasets. As shown in Table II, results on small-scale datasets are usually better with text embedding. We hypothesise this is because insufficient audio training data leads to sub-optimal learning of generative models, such as overfitting. This hypothesis is supported by our result, which shows that training models with audio embedding achieve the best performance with fewer training steps, such as 20k steps in US8K and 80k steps in AudioCaps. Conversely, texts or labels provide less detailed and diverse conditions, which can regularize the model to learn data distribution with less chance of overfitting, leading to model convergence with more training steps at the same time.

Figure 2 illustrates the performance of AudioLDM on different training epochs with and without pertaining and different modalities as condition information. The experiment is performed on the ESC50 dataset. We notice that 1) the pre-trained model can reach coverage quickly with text-embedding, with about 20k training steps; 2) AudioLDM can achieve better performance with text-embedding on ESC50; 3) AudioLDM trained from scratch converge much slower and may not converge well, even with a larger number of steps.

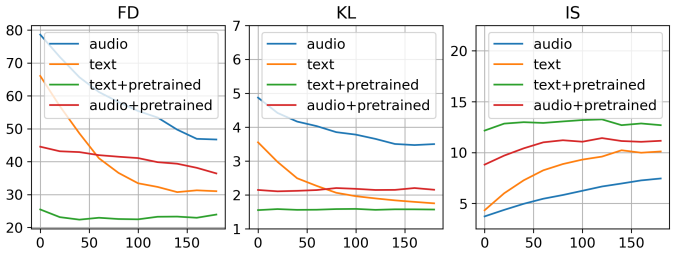

Fig. 2. The performance of AudioLDM on ESC50 as a function of thousand training steps. Four curves show AudioLDM optimized with 1) audio embeddings; 2) text embeddings; 3) text embedding with pre-trained parameters; and 4) audio embedding with pre-trained parameters

V. CONCLUSION

This work investigates the effect of pre-trained AudioLDM on sound generation tasks, with results on various settings and datasets. This study shows the pre-trained audio generative model can improve the sample quality and reduce the training time, especially with smaller-scale datasets. This serves as evidence for future studies on audio generation in data-scarcity scenarios. Besides, we conclude that text embedding is preferred as the condition information on small-scale datasets, with the effect of alleviating overfitting during training. Finally, a new benchmark is established for sound generation tasks with four commonly used datasets. These baseline results can be used as a reference for future studies of sound generation.

VI. Acknowledgment

This research was partly supported by a research scholarship from the China Scholarship Council (CSC) No.202208060240, the British Broadcasting Corporation Research and Development (BBC R&D), Engineering and Physical Sciences Research Council (EPSRC) Grant EP/T019751/1 "AI for Sound", and a PhD scholarship from the Centre for Vision, Speech and Signal Processing (CVSSP), University of Surrey. For the purpose of open access, the authors have applied a Creative Commons Attribution (CC BY) license to any Author Accepted Manuscript version arising.

References

[1] X. Liu, T. Iqbal, J. Zhao, Q. Huang, M. Plumbley, and W. Wang, "Conditional sound generation using neural discrete time-frequency representation learning," *IEEE International Workshop on Machine Learning for Signal Processing*, pp. 1–6, 2021.

[2] F. Kreuk, G. Synnaeve, A. Polyak, U. Singer, A. Défossez, J. Copet, D. Parikh, Y. Taigman, and Y. Adi, "AudioGen: Textually Guided Audio Generation," *arXiv preprint arXiv:2209.15352*, 2022.

[3] X. Tan, J. Chen, H. Liu, J. Cong, C. Zhang, Y. Liu, X. Wang, Y. Leng, Y. Yi, L. He *et al.*, "Naturalspeech: End-to-end text to speech synthesis with human-level quality," *arXiv preprint arXiv:2205.04421*, 2022.

[4] V. Iashin and E. Rahtu, "Taming Visually Guided Sound Generation," *arXiv preprint arXiv:2110.08791*, 2021.

[5] X. Liu, H. Liu, Q. Kong, X. Mei, J. Zhao, Q. Huang, M. D. Plumbley, and W. Wang, "Separate what you describe: Language-queried audio source separation," *arXiv preprint arXiv:2203.15147*, 2022.

[6] S. Ghose and J. J. Prevost, "AutoFoley: Artificial Synthesis of Synchronized Sound Tracks for Silent Videos with Deep Learning," *arXiv preprint arXiv:2002.10981*, 2020.

[7] H. Liu, W. Choi, X. Liu, Q. Kong, Q. Tian, and D. Wang, "Neural Vocoder is All You Need for Speech Super-resolution," *arXiv preprint arXiv:2203.14941*, 2022.

[8] D. Yang, J. Yu, H. Wang, W. Wang, C. Weng, Y. Zou, and D. Yu, "Diffsound: Discrete Diffusion Model for Text-to-sound Generation," *arXiv preprint arXiv:2207.09983*, 2022.

[9] H. Liu, Z. Chen, Y. Yuan, X. Mei, X. Liu, D. Mandic, W. Wang, and M. D. Plumbley, "AudioLDM: Text-to-Audio Generation with Latent Diffusion Models," *arXiv preprint arXiv:2301.12503*, 2023.

[10] R. Huang, J. Huang, D. Yang, Y. Ren, L. Liu, M. Li, Z. Ye, J. Liu, X. Yin, and Z. Zhao, "Make-An-Audio: Text-To-Audio Generation with Prompt-Enhanced Diffusion Models," *arXiv preprint arXiv:2301.12661*, 2023.

[11] D. Hendrycks, K. Lee, and M. Mazeika, "Using pre-training can improve model robustness and uncertainty," in *International Conference on Machine Learning*, vol. 97, 2019, pp. 2712–2721.

[12] D. Erhan, A. Courville, Y. Bengio, and P. Vincent, "Why does unsupervised pre-training help deep learning?" in *Proceedings of the International Conference on Artificial Intelligence and Statistics*, vol. 9, 2010, pp. 201–208.

[13] Z. Guo, M. Yan, J. Qi, J. Zhou, Z. He, Z. Lin, G. Zheng, and X. Wang, "Few-Shot Table-to-Text Generation with Prompt Planning and Knowledge Memorization," *arXiv preprint arXiv:2302.04415*, 2023.

[14] X. Liu, X. Mei, Q. Huang, J. Sun, J. Zhao, H. Liu, M. D. Plumbley, V. Kilic, and W. Wang, "Leveraging pre-trained bert for audio captioning," in *European Signal Processing Conference*, 2022, pp. 1145–1149.

[15] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension," *arXiv preprint arXiv:1910.13461*, 2019.

[16] B. Li, X. Wang, X. Xu, Y. Hou, Y. Feng, F. Wang, and W. Che, "Semantic-Guided Image Augmentation with Pre-trained Models," *arXiv preprint arXiv:2302.02070*, 2023.

[17] Y. Zhang, S.-C. Huang, Z. Zhou, M. P. Lungren, and S. Yeung, "Adapting Pre-trained Vision Transformers from 2D to 3D through Weight Inflation Improves Medical Image Segmentation," *arXiv preprint arXiv:2302.04303*, 2023.

[18] A. Ghanbarzade and H. Soleimani, "Self-Supervised In-Domain Representation Learning for Remote Sensing Image Scene Classification," *arXiv preprint arXiv:2302.01793*, 2023.

[19] X. Mei, X. Liu, H. Liu, J. Sun, M. D. Plumbley, and W. Wang, "Language-based audio retrieval with pre-trained models," Tech. Rep., DCASE Challenge, 2022.

[20] 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.

[21] C. D. Kim, B. Kim, H. Lee, and G. Kim, "AudioCaps: Generating captions for audios in the wild," in *NAACL-HLT*, 2019.

[22] J. F. Gemmeke, D. P. W. Ellis, D. Freedman, A. Jansen, W. Lawrence, R. C. Moore, M. Plakal, and M. Ritter, "AudioSet: An ontology and human-labeled dataset for audio events," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017, pp. 776–780.

[23] J. Salamon, C. Jacoby, and J. P. Bello, "A dataset and taxonomy for urban sound research," in *ACM International Conference on Multimedia*, 2014, pp. 1041–1044.

[24] K. J. Piczak, "ESC: Dataset for Environmental Sound Classification," in *Proceedings of the Annual ACM Conference on Multimedia*, 2015, pp. 1015–1018.

[25] T. Salimans, I. Goodfellow, W. Zaremba, V. Cheung, A. Radford, and X. Chen, "Improved Techniques for Training GANs," *arXiv preprint arXiv:1606.03498*, 2016.

[26] K. Kilgour, M. Zuluaga, D. Roblek, and M. Sharifi, "Fréchet Audio Distance: A Metric for Evaluating Music Enhancement Algorithms," *arXiv preprint arXiv:1812.08466*, 2018.

[27] Q. Kong, Y. Xu, I. Iqbal, Y. Cao, W. Wang, and M. Plumbley, "Acoustic scene generation with conditional samplernn," *IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 925–929, 2019.

[28] A. van den Oord, O. Vinyals, and K. Kavukcuoglu, "Neural Discrete Representation Learning," *arXiv preprint arXiv:1711.00937*, 2017.

[29] A. Radford, J. W. Kim, C. Hallacy, A. Ramesh, G. Goh, S. Agarwal, G. Sastry, A. Askell, P. Mishkin, J. Clark *et al.*, "Learning transferable visual models from natural language supervision," in *International Conference on Machine Learning*, 2021, pp. 8748–8763.

[30] 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," *arXiv preprint arXiv:1910.10683*, 2019.

[31] M. Heusel, H. Ramsauer, T. Unterthiner, B. Nessler, and S. Hochreiter, "GANs Trained by a Two Time-Scale Update Rule Converge to a Local Nash Equilibrium," *arXiv preprint arXiv:1706.08500*, 2017.

[32] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Conference on Neural Information Processing Systems*, 2020.

[33] K. Simonyan and A. Zisserman, "Very Deep Convolutional Networks for Large-Scale Image Recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[34] 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," *arXiv preprint arXiv:1912.10211*, 2019.