Synthesizing Soundscapes: Leveraging Text-to-Audio Models for Environmental Sound Classification

Francesca Ronchini, Luca Comanducci, Fabio Antonacci
Dipartimento di Elettronica, Informazione e Bioingegneria (DEIB), Politecnico di Milano
Piazza Leonardo Da Vinci 32, 20133 Milan, Italy
francesca.ronchini@polimi.it, luca.comanducci@polimi.it, fabio.antonacci@polimi.it

***Abstract*—In the past few years, text-to-audio models have emerged as a significant advancement in automatic audio generation. Although they represent impressive technological progress, the effectiveness of their use in the development of audio applications remains uncertain. This paper aims to investigate these aspects, specifically focusing on the task of classification of environmental sounds. This study analyzes the performance of two different environmental classification systems when data generated from text-to-audio models is used for training. Two cases are considered: a) when the training dataset is augmented by data coming from two different text-to-audio models; and b) when the training dataset consists solely of synthetic audio generated. In both cases, the performance of the classification task is tested on real data. Results indicate that text-to-audio models are effective for dataset augmentation, whereas the performance of the models drops when relying on only generated audio.**

***Index Terms*—audio generation, machine learning, data augmentation, text-to-audio generative models**

I. Introduction

In recent years, the research field related to sound synthesis has seen the introduction of Text-to-Audio (TTA) models. TTA models are sound generative systems that use natural language prompts to guide the generation of the desired audio samples [1]–[4]. They offer greater flexibility compared to alternative sound synthesis approaches that use one-hot encoding to condition the generation of sound samples according to the desired sound class [5].

TTA systems represent a significant advancement in the automatic audio generation and several models have been presented with the ability to generate considerably high-quality general audio [1]–[4]. Kreuk et al. proposed AudioGen [2], which learns a discrete latent representation from raw waveforms, while an attention-based autoregressive decoder is used to generate a discrete latent stream. Yang et al. introduced DiffSound [1], a text-to-sound generative model based on a non-autoregressive token-decoder based on a discrete diffusion model. Liu et al. proposed AudioLDM [3], which is based on a latent diffusion model [6] that allows the extension from learning discrete representations, as previously done in TTA approaches, to continuous latent representations. The same authors proposed AudioLDM2 [4], a more sophisticated version of AudioLDM. Other examples of TTA systems are Tango [7], Make-an-Audio [8], and Audiobox [9].

TTA models can find applications in different domains, such as augmented or virtual reality [10], and foley sound generation [11], among others. Thanks to their versatility in generating audio signals across a broad range of content types, they also offer significant potential for dataset synthesis or augmentation across different tasks. Their ability to generate high-quality general sounds, together with their flexibility and ease of utilization, make them interesting tools for generating datasets that could be directly integrated into the training pipeline of deep learning systems. Nevertheless, to the best of our knowledge, limited literature is available regarding the utilization of TTA-generated datasets for training deep learning models tailored to specific tasks. In [12], Kroher et al. trained a music genre classifier on a fully artificial music dataset generated with MusicGen [13], a text-to-music generation model. In [14], the authors fine-tune a TTA model for generating anomalous sounds for the Anomalous Sound Detection task.

Motivated by positive preliminary findings from [12] and [14], this paper aims to investigate how to leverage TTA systems in the field of Detection and Classification of Acoustic Scenes and Events (DCASE) [15]. The main goal of the DCASE field is the automatic recognition of sound events and scenes and it can be divided into different tasks: Sound Event Detection [16], Sound Event Localization [17], Acoustic Scene Classification [18], among others. This domain presents itself as an ideal application area to investigate the possibility of using training datasets generated via TTAs, which have the ability to reproduce sounds present in most DCASE-related datasets. To analyze how to exploit the TTA models in the DCASE domain, we selected the Environmental Sound Classification (ESC) task [19]. ESC is a well-known task in the DCASE community and different approaches have been proposed in the literature [19]. In [20], Salamon and Bello proposed a deep learning method, using the UrbanSound8K (US8K) [21] dataset. US8K became a popular dataset for ESC and, nowadays, it is a state-of-the-art benchmark for sound classification tasks. ESC can be considered one of the base case scenarios for DCASE. As such, we choose it in this study to follow a bottom-up approach in terms of complexity. We first analyze a simpler problem, to then generalize the analysis and techniques to more complex tasks in future works. The same reasoning applies to the choice of the US8K dataset.

To the best of our knowledge, this is the first paper to analyze how to leverage TTA-generated datasets into the training pipeline of ESC systems. We considered two scenarios: **a) TTA-augmented training dataset**: the training dataset is augmented with data generated with TTA models; **b) TTA-generated training dataset**: the training dataset is composed of only synthetic audio generated through TTA models. We selected two TTA models. With each of them, we generate two versions of the US8K dataset. For one version we used a single-instruction text prompt, and for the the other one we used a text prompt proposed by a Large Language Model (LLM) [22]. We then selected two deep-learning ESC models of differing architectural complexity, and trained them using datasets augmented or generated through TTA, depending on the scenario. We then evaluated how the accuracy performance of both ESC models was influenced by testing them on real-world data. Audio samples and the code used to obtain the results are available at https://ronfrancesca.github.io/Text-to-Audio-ESC/.

II. Related background

A. Synthetic Data in Environmental Sound Classification

Synthetic data are beneficial for ESC applications. In fact, one limitation of deep learning solutions for ESC is the need of large amounts of labeled training data to reach good performances, and labeling a dataset is both time-consuming and biases-prone [23]. Different studies have demonstrated that incorporating synthetic data alongside real-world recordings during the training phase enhances system generalization and leads to improved performances [24]–[26]. The synthetic data considered in [24]–[26] were generated using available tools for synthesizing soundscapes. A notable example of these is Scaper [24], an open-source Python library, which is among the main libraries used for soundscape synthesis and augmentation in classification tasks. It allows the creation of new soundscapes or the integration of sound events into pre-existing ones, managing different parameters (number of sound events, types, etc.) [24]. Nevertheless, these parameters need to be specified beforehand, and a pre-organized and available sound collection is necessary for soundscape creation. The TTA models considered in this paper could be a potential solution to these limitations. They provide the ability to customize and specify the context of the desired audio output using natural language, enabling precise control with text prompts over the generated sounds in an easy and user-friendly way. Additionally, given that collecting real data is an expensive operation, employing TTA models for training allows to preserve real data for testing purposes, especially in cases where there is scarcity of them.

B. Text-to-Audio generative models

This section briefly introduces the two TTA models used to generate synthetic data for this study. AudioGen [2] is an auto-regressive model for TTA based on two steps: 1) Learning a discrete representation of the raw audio through

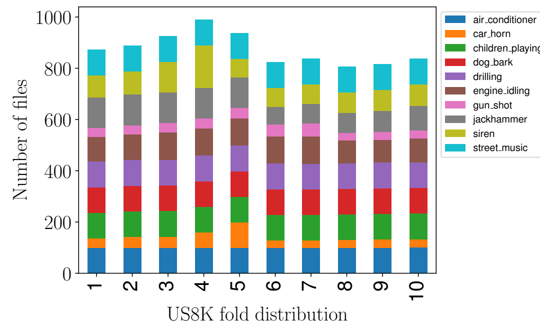


Fig. 1: US8K dataset classes distribution per each fold. Colors represent the different sound classes, specified in the legend.

an auto-encoding procedure; 2) Generating the audio through a transformer model applied over the learned codes and conditioned through textual features. The audio tokenization procedure is performed via an auto-encoder model trained using a GAN-like objective and a multi-scale STFT discriminator. The decoding procedure is performed via a transformer model where text embeddings are used for conditioning. AudioLDM [3] is a TTA system based on a continuous latent-diffusion model conditioned via CLAP [27]. This removes the need of paired audio-text data during the training process. AudioLDM2 [4] extends this approach introducing a general audio representation denoted as *Language of Audio (LOA)* used to extract conditioning embeddings by training an Audio Masked AutoEncoder (AudioMAE) [28]. This enables the generation of any audio with greater control and flexibility.

III. Experimental Setup

A. Text-to-Audio models

Both AudioGen [2] and AudioLDM2 [4] are open-source and the authors released pre-trained models that can be readily used to generate audio. For AudioGen, we used the available API for the pre-trained model *audiogen-medium*, composed of 1.5 B of parameters. For AudioLDM2, we used the *audioldm* checkpoint, whose model size is 1.1 B parameters. We focused on the smaller version of the model, which is faster in terms of generation. For each class, we considered 200 inference steps and a seed $= 20$. This is valid only for the AudioLDM2 since it is based on a diffusion model. For both models, the audio length is of 4 s (the maximum length of audio files in US8K).

B. Datasets generation

US8K [21] is the reference dataset selected for this study. It contains 8732 labeled sounds of 4 s maximum duration of urban sounds from 10 classes (reported in Fig. 1). The dataset is divided into 10 folds, used for leave-one-out cross-validation at evaluation time. The distribution of the sounds in the different folds is reported in Fig. 1.

We generated four versions of the US8K dataset: two with AudioLDM2 and two with AudioGen. For each TTA model, we used two different prompt templates (better described in Section III-C). For each dataset version, we first generated the total amount of data following the class distributions of US8K.

We then divided the generated dataset into 10 folds, following the same partition proposed for US8K.

C. Prompt templates

We use two different prompt templates for each TTA model. The first consists of a single-instruction sentence containing the instruction of a clear sound generation in a urban context, specifying the desired audio class. The template is: *"A clear sound of a <class_to_generate> in a urban context"*. (E.g. for the *dog_bark* class the template is *"A clear sound of a dog barking"*). For AudioLDM2, as suggested by the authors on the available guideline, we used a negative prompt to reach a better quality of the output. In this case the negative prompt is: *"Low quality"*. AudioGen does not involve the use of a negative prompt.

The second template has been generated using a Large Language Model (LLM), ChatGPT 3.5. The LLM has been asked to generate a single and clear sentence that would have been used as input for a TTA model to generate an audio sample of a sound class in a urban context. The prompt template suggested by ChatGPT is: *"Generate a realistic audio representation of the sound of a <class_to_generate> in a urban environment"*. For AudioLDM2, also the negative prompt has been asked to ChatGPT. Depending on the sound class, the templates have been adapted to have repetitive sounds to be consistent with the padding strategy implemented in the study (e.g dog_bark or car_horn), or to better specify a sound coming from a class which might induct confusion on the generation of its sound (e.g. siren).

D. Baselines and configuration

To better generalize the results, we implemented two ESC models. The first classifier is a CNN implemented following the same parameters and similar structure of the one in [20]. Our implementation is slightly different so, as is common in practice, results will not be exactly the same as the original paper. The CNN implemented for this study is composed of three convolutional layers, each followed by a max-pooling operation, except the last layer. The kernel size is 5 and the max pooling operation implies a stride of 4 for the time dimension and 2 for the frequency dimension, respectively. The last two layers are fully connected layers, each preceded by a dropout of 0.5. The second classifier is a state-of-the-art CRNN. We followed a similar structure of the CRNN proposed in [29], inspired by [30]. The network is composed of seven convolutional blocks followed by a bidirectional GRU layer and a dense layer that generates the final output. We used the same parameters and configuration proposed in [29].

For both networks, the input are TF patches of 3 s taken from the log mel-spectrogram computed from the audio input, as in [20]. All the sounds of US8K have been resampled to 16 kHz, being this the frequency at which the TTA models selected for the study generate sounds. We computed the STFT considering a Hann window of 1024 samples, and 2048 frequency points. We considered 64 mel-bands for the log mel-spectrogram with a frequency range between 0 Hz and 8000 Hz. Both networks have been trained for 100 epochs, with batch size of 128 and an early stop condition with patience on the validation loss of 15 epochs. We considered Adam optimizer with a learning rate of 0.001. Samples shorter than 4 s have been padded by repeating the sample until reaching the desired time length.

TABLE I: Data augmentation comparison

Data aug. method	Accuracy (CNN)	Accuracy (CRNN)
US8K-PS	66.49 (0.60)	65.01 (0.95)
US8K-TS	64.14 (0.80)	62.63 (1.80)
US8K-AudioGen	68.42 (0.71)	65.18 (0.87)
US8K-AudioGen$_{\text{gpt}}$	68.88 (0.50)	**65.39 (0.63)**
US8K-AudioLDM2	68.04 (0.63)	63.41 (0.99)
US8K-AudioLDM2$_{\text{gpt}}$	**69.64 (0.91)**	64.69 (0.53)
US8K (Baseline)	64.68 (0.82)	62.70 (0.65)

E. Experiments definition

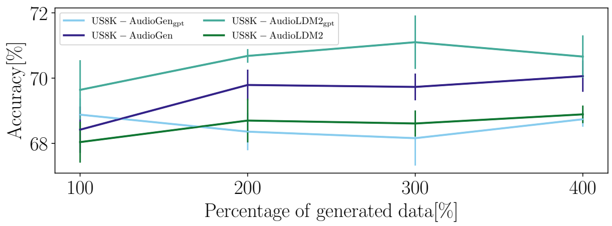
To understand the impact of the TTA-generated dataset during the training of ESC learning-based models, we consider two cases: **TTA-augmented training dataset** and **TTA-generated training dataset**.

For the **TTA-augmented training dataset** case, we trained the two networks with the US8K dataset by augmenting it with one version of the TTA-generated datasets. The TTA-generate datasets and US8K have the same distributions, so this can be considered as a data augmentation technique where the original dataset is incremented by 100 %. We compared the results with two signal processing data augmentation techniques proposed in [20]: Time Stretching (TS) and Pitch Shifting (PS). TS is the process of changing the speed of an audio signal without affecting its pitch, while PS is the process of changing the pitch without affecting the speed of the audio sample. We focused on PS1 (as proposed in [20]), here referred as PS. In [20], the authors augmented each audio file four times, using four values both for TS and PS. In contrast, for each file we randomly select only one value to double the USK8 size. Encouraged by the outcomes of this experiment, we focus on TTA-augmented techniques and investigate whether proportionally expanding the dataset size will reflect the improvement in model accuracy.

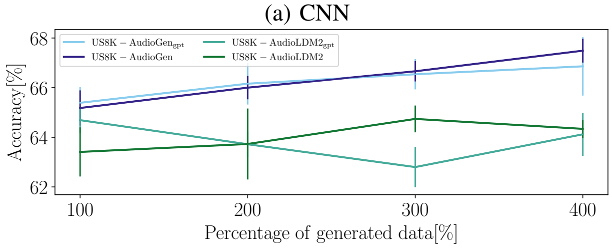
For the **TTA-generated training dataset** case, we trained the ESC models only with TTA-generated versions of US8K, one by one, and analyzed the impact on the model's accuracy. In both scenarios, we compared the results of the experiments with the ESC models trained with only the original US8K. In all cases, we evaluated the systems using leave-one-out cross-validation [20]. We used 9 folds for training (8 for training and 1 for validation), and 1 fold for testing. The testing fold was always selected from US8K, to evaluate the ESC models on real-world data and make the comparison as fair as possible.

IV. Experiments results

This section reports the results of the two experiments described in Sec. III-E. In both scenarios, *AudioGen* and *AudioLDM2* indicate the dataset versions generated with AudioGen



(a) CNN



(b) CRNN

Fig. 2: Classification accuracy when varying the size of the TTA-generated augmentation dataset. Error bars represent 95% confidence intervals over 5 repetitions of the experiment.

and AudioDM2, respectively. In case of a **US8K-** prefix, it means they have been added to the US8K dataset at training. The *gpt* subscript indicates that the text prompts for the TTA models have been generated using ChatGPT, otherwise with the single-instruction prompt. The results report the accuracy average over 5 runs of the experiments, with the related 95% confidence interval (in parenthesis).

A. TTA-augmented training dataset case

In this scenario, we aim to answer the question: *Does the integration of TTA-generated audio samples as a data augmentation technique affect the accuracy of ESC systems?* Table I reports the accuracy results for the different data augmentation techniques considered for both the ESC approaches. *USK-PS* and *USK-TS* indicate that PS and TS data augmentation has been applied to the US8K dataset, respectively. The last row reports the accuracy for the CNN and CRNN trained only with the original US8K dataset.

From the results, it is possible to observe that almost all the TTA-based augmentation techniques reach higher performances compared to the signal processing ones. Training models using GPT-based dataset versions evidentiates this, especially in the case of the CNN model, which yields its optimal performance when augmented with *AudioLDM2$_{gpt}$*, achieving nearly a 5% increase in accuracy over the baseline. For the CRNN model, the most favorable results are obtained using *AudioGen$_{gpt}$*, exhibiting a 3% enhancement compared to the baseline. Data augmentation techniques based on signal processing consistently yield inferior performance for the CNN architecture, while getting comparable performances in the case of CRNN. These findings suggest that incorporating TTA-generated audio samples as a data augmentation technique enhances the performance of the ESC system.

TABLE II: Text-to-Audio dataset comparison

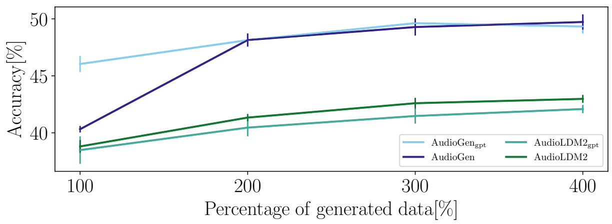
Training dataset	Accuracy (CNN)	Accuracy (CRNN)
AudioGen	40.32 (0.29)	38.79 (1.24)
AudioGen$_{gpt}$	**46.04 (0.71)**	**43.96 (1.36)**
AudioLDM2	38.81 (0.56)	36.11 (1.11)
AudioLDM2$_{gpt}$	38.49 (1.21)	32.86 (1.01)
US8K (Baseline)	64.68 (0.82)	62.70 (0.65)

Motivated by these results we perform a further experiment to understand if increasing the size of the TTA-generated dataset corresponds to an equivalent increase in performance. Specifically, we consecutively double the size of the data used for augmentation, up to 400% the original size. We increased the size of the dataset always following the same distribution of US8K. The case of 100% corresponds to the previous experiment. The accuracy when varying the size of the TTA-generated dataset is reported in Fig. 2(a) for the CNN, and in Fig. 2(b) for the CRNN. No clear trend is observed for both models. Only for the CRNN it is possible to notice an improvement on the performances when the US8K dataset is augmented with up to 200% - 300% of data coming from either one of the two AudioGen generated versions of it. However, these results underscore the necessity for further investigation to understand the implications of the observed patterns.

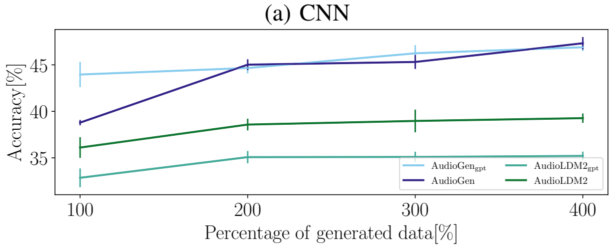
B. TTA-generated training dataset

This case focuses on answering the question *Can we rely on only TTA-generated data to train an ESC system?* Table II reports the accuracy of the two ESC models when trained with one of the TTA-generated versions of the US8K dataset.

The results show that neither one of the two ESC models trained with only TTA-generated datasets reaches the baselines performances. However, the highest accuracy for both models is reached when trained with GPT-based dataset versions generated with AudioGen. This confirms that AudioGen is preferable as TTA model and allows to get higher performances when used as dataset generator for ESC. On the contrary, when using AudioLDM2 performances worsen. At any rate, the results reveal that is not possible to rely on only TTA-generated dataset yet. Our intuition is that domain adaptation between the TTA-data used during training and the USK8 real data used for testing has an impact on the performances. As for the previous scenario, we wondered if the threshold for achieving baseline performance might be influenced by the quantity of data used at training. Therefore, we incrementally increased the dataset size incrementally doubling it, extending it to train models with up to 400% of synthetic data. Fig. 3(a) shows the performances for the CNN model; Fig. 3(b) reports the performances for the CRNN model. As we can observe, increasing the number of audio data is useful up to 2-3 times the original dataset size, confirming the results of the previous case experiments. Also in this case, both networks achieve higher performances when trained with AudioGen dataset versions. This underscores the AudioGen



(a) CNN



(b) CRNN

Fig. 3: Classification accuracy when varying the size of the training dataset composed of only TTA-generated data. Error bars: 95% confidence intervals over 5 experiment repetitions.

ability to generate more realistic sounds, enabling the ESC models to better generalize on real-world data.

V. Conclusions and Future Works

In this paper, we analyzed the impact of including Text-To-Audio-generated datasets in the training process of learning-based models for ESC. We examined two ESC models and two TTA models across two scenarios: a) augmenting the training dataset with data generated using TTA models; b) the training dataset solely consists of synthetic audio generated by TTA models. From the results, we can conclude that TTA-generated datasets are beneficial when used as data augmentation techniques, but are not ready to be used as the only source of data during training. We believe that the obtained results motivate further investigations on the topic. Future works will include the exploration of more advanced prompt engineering strategies and the investigation of fine-tuning methods to improve generation capabilities of TTA models. We will further investigate with training the models with simultaneously artificial data and real-world data, to find a balance between them which allows to overcome the expensive operation of collecting and labeling real-world data.

References

[1] D. Yang, J. Yu, H. Wang, W. Wang, C. Weng, Y. Zou, and D. Yu, "Diffsound: Discrete diffusion model for text-to-sound generation," *IEEE/ACM Trans. Audio Speech Lang. Process.*, 2023.

[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," in *ICLR*, 2022.

[3] 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:2301.12503*, 2023.

[4] H. Liu, Q. Tian, Y. Yuan, X. Liu, X. Mei, Q. Kong, Y. Wang, W. Wang, Y. Wang, and M. D. Plumbley, "Audioldm 2: Learning holistic audio generation with self-supervised pretraining," *arXiv:2308.05734*, 2023.

[5] X. Liu, T. Iqbal, J. Zhao, Q. Huang, M. D. Plumbley, and W. Wang, "Conditional sound generation using neural discrete time-frequency representation learning," in *Proc. MLSP*. IEEE, 2021.

[6] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proc. IEEE/CVF CVPR*, 2022.

[7] D. Ghosal, N. Majumder, A. Mehrish, and S. Poria, "Text-to-audio generation using instruction-tuned llm and latent diffusion model," *arXiv:2304.13731*, 2023.

[8] 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:2301.12661*, 2023.

[9] A. Vyas, B. Shi, M. Le, A. Tjandra, Y.-C. Wu, B. Guo, J. Zhang, X. Zhang, R. Adkins, W. Ngan *et al.*, "Audiobox: Unified audio generation with natural language prompts," *arXiv:2312.15821*, 2023.

[10] R. Nordahl, L. Turchet, and S. Serafin, "Sound synthesis and evaluation of interactive footsteps and environmental sounds rendering for virtual reality applications," *Trans. Vis. Comput. Graph.*, 2011.

[11] Y. Chung, J. Lee, and J. Nam, "T-foley: A controllable waveform-domain diffusion model for temporal-event-guided foley sound synthesis," *arXiv:2401.09294*, 2024.

[12] N. Kroher, H. Cuesta, and A. Pikrakis, "Can musicgen create training data for mir tasks?" *arXiv:2311.09094*, 2023.

[13] J. Copet, F. Kreuk, I. Gat, T. Remez, D. Kant, G. Synnaeve, Y. Adi, and A. Défossez, "Simple and controllable music generation," *NeurIPS*, 2024.

[14] H. Zhang, Q. Zhu, J. Guan, H. Liu, F. Xiao, J. Tian, X. Mei, X. Liu, and W. Wang, "First-shot unsupervised anomalous sound detection with unknown anomalies estimated by metadata-assisted audio generation," *arXiv:2310.14173*, 2023.

[15] T. Virtanen, M. D. Plumbley, and D. Ellis, *Computational analysis of sound scenes and events*. Springer, 2018.

[16] A. Mesaros, T. Heittola, T. Virtanen, and M. D. Plumbley, "Sound event detection: A tutorial," *IEEE Signal Processing Magazine*, 2021.

[17] A. Politis, A. Mesaros, S. Adavanne, T. Heittola, and T. Virtanen, "Overview and evaluation of sound event localization and detection in dcase 2019," *IEEE/ACM Trans. Audio Speech Lang. Process.*, 2020.

[18] D. Barchiesi, D. Giannoulis, D. Stowell, and M. D. Plumbley, "Acoustic scene classification: Classifying environments from the sounds they produce," *IEEE Signal Processing Magazine*, 2015.

[19] A. Bansal and N. K. Garg, "Environmental sound classification: A descriptive review of the literature," *ISWA*, 2022.

[20] J. Salamon and J. P. Bello, "Deep convolutional neural networks and data augmentation for environmental sound classification," *IEEE Signal Process. Lett.*, 2017.

[21] J. Salamon, C. Jacoby, and J. P. Bello, "A dataset and taxonomy for urban sound research," in *Proc. ACM Multimed.*, 2014.

[22] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell *et al.*, "Language models are few-shot learners," *NeurIPS*, 2020.

[23] F. Ronchini, R. Serizel, N. Turpault, and S. Cornell, "The impact of non-target events in synthetic soundscapes for sound event detection," in *DCASE Workshop*, 2021.

[24] J. Salamon, D. MacConnell, M. Cartwright, P. Li, and J. P. Bello, "Scaper: A library for soundscape synthesis and augmentation," in *Proc. WASPAA*. IEEE, 2017.

[25] F. Ronchini and R. Serizel, "A benchmark of state-of-the-art sound event detection systems evaluated on synthetic soundscapes," in *Proc. ICASSP*. IEEE, 2022.

[26] N. Turpault, R. Serizel, A. P. Shah, and J. Salamon, "Sound event detection in domestic environments with weakly labeled data and soundscape synthesis," in *DCASE workshop*, 2019.

[27] 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.

[28] P.-Y. Huang, H. Xu, J. Li, A. Baevski, M. Auli, W. Galuba, F. Metze, and C. Feichtenhofer, "Masked autoencoders that listen," *Advances in Neural Information Processing Systems*, 2022.

[29] C. Castorena, M. Cobos, J. Lopez-Ballester, and F. J. Ferri, "A safety-oriented framework for sound event detection in driving scenarios," *Applied Acoustics*, 2023.

[30] J. Sang, S. Park, and J. Lee, "Convolutional recurrent neural networks for urban sound classification using raw waveforms," in *26th European Signal Processing Conference (EUSIPCO)*. IEEE, 2018.