AudioLDM 2: Learning Holistic Audio Generation with Self-supervised Pretraining

Haohe Liu, Yi Yuan, Xubo Liu, Xinhao Mei, Qiuqiang Kong,
Qiao Tian, Yuping Wang, Wenwu Wang, Yuxuan Wang, Mark D. Plumbley

Abstract—Although audio generation shares commonalities across different types of audio, such as speech, music, and sound effects, designing models for each type requires careful consideration of specific objectives and biases that can significantly differ from those of other types. To bring us closer to a unified perspective of audio generation, this paper proposes a holistic framework that utilizes the same learning method for speech, music, and sound effect generation. Our framework utilizes a general representation of audio, called "language of audio" (LOA). Any audio can be translated into LOA based on AudioMAE, a self-supervised pre-trained representation learning model. In the generation process, we translate other modalities into LOA by using a GPT-2 model, and we perform self-supervised audio generation learning with a latent diffusion model conditioned on the LOA of audio in our training set. The proposed framework naturally brings advantages such as reusable self-supervised pretrained latent diffusion models. Experiments on the major benchmarks of text-to-audio, text-to-music, and text-to-speech with three *AudioLDM 2* variants demonstrate competitive performance of the *AudioLDM 2* framework against previous approaches. Our code, pretrained model, and demo are available at https://audioldm.github.io/audioldm2.

Index Terms—audio generation, diffusion model, self-supervised learning, speech synthesis, AIGC

I. Introduction

ARTIFICIAL intelligence generated content (AIGC) refers to any digital content such as images, videos, text, or audio that has been fully or partially created by an AI system without human involvement in the creative process [1]. Of particular interest is the ability of AI to produce audio content based on text, phonemes, or images [2]–[4]. AI-based audio generation has a wide potential in applications including synthesizing human or artificial voices for digital assistants [5], generating sound effects and background music for movies, and games [6], and automating the production of podcasts and audiobooks [7].

AI-based audio generation is often undertaken in separate sub-domains, such as the generation of speech [2], music [8], sound effects [4], and specific types of sounds such as footsteps and violin sounds [9], [10]. To address the specific challenges in each sub-domain, most previous works design task-specific inductive biases, which are predefined constraints that guide the learning process to a specific problem space. For example, pitch and duration predictors are often used in speech synthesis to model the prosody of speech [2], [11], while MIDI representation [12] and domain-specific pre-trained modules are often used in music generation [8], [13].

Despite significant progress being made in developing specialized models for specific sub-domains of audio generation, the limitations of such specialization restrict the broader application of audio-generation models in complex auditory scenarios. Although there are models that can generate various types of audio, such as AudioLDM [4], the speech they generate is still not intelligible. Whether a unified approach can be developed to generate various types of audio signals, including intelligible speech, remains unanswered. Different types of sound can occur simultaneously in real-world cases, such as in movie scenes, requiring a more general approach to modelling audio generation. While there are works that address audio generation in a general domain, they mostly focus on generating audio with correct audio events with limited attention to detail. For example, previous text-to-audio generation research tends to generate unintelligible speech [4], [14], [15]. Moreover, while inductive biases have been useful in addressing the challenges of specific sub-domains, conclusions about a specific design drawn from one domain may not necessarily transfer to another. Recent advancements in addressing problems from a unified perspective have yielded substantial progress [16]–[19]. This trend highlights the potential of constructing a unified audio generation framework.

This paper presents a novel and versatile framework, called *AudioLDM 2*, that can generate audio with flexible conditions, without the need for domain-specific inductive bias. The core idea is to introduce a sequence of vectors that represent the semantic information of an audio clip, which we will refer to as the "language of audio" (LOA). This approach allows us to translate human-understandable information into LOA and synthesize audio representation conditioned on LOA. This idea is similar to the use of onomatopoeia in [20] to describe environmental sounds. However, although onomatopoeia can effectively mimic certain sounds like animal noises or simple actions (e.g., "splash" for water), it can not encompass the full range of audio nuances. In theory, the "language of audio" should be able to represent both fine-grained acoustic information (e.g., "what does the speaker say") and coarse-grained semantic information (e.g., "what is that sound"). Considering these requirements, we propose to leverage the

Haohe Liu, Yi Yuan, Xubo Liu, Xinhao Mei, Wenwu Wang, and Mark D. Plumbley are with the Centre for Vision, Speech and Signal Processing (CVSSP), University of Surrey, Guilford, UK. Email: {haohe.liu, yi.yuan, xubo.liu, x.mei, w.wang, m.plumbley}@surrey.ac.uk.

Qiuqiang Kong is with the Department of Electronic Engineering, Chinese University of Hong Kong, Hong Kong, China. Email: qqkong@ee.cuhk.edu.hk.

Qiao Tian, Yuping Wang and Yuxuan Wang: are with the Speech, Audio & Music Intelligence (SAMI) Group, ByteDance Inc. Email: {tianqiao.wave, kongqiuqiang, wangyuping, wangyuxuan.11}@bytedance.com.

features extracted by an audio masked autoencoder (AudioMAE) [21], an audio-generative self-supervised pretraining framework. An AudioMAE is pre-trained on diverse audio content, and its dual generative and reconstructive pre-training approach makes it potentially a strong option for representing audio in generative tasks.

Specifically, we utilize a GPT-2 language model [22] to translate conditioning information into the AudioMAE features. We then use a latent diffusion model [23] to synthesize audio based on the AudioMAE features. The latent diffusion model can be optimized in a self-supervised manner, allowing for pre-training with large-scale unlabelled audio data. Our language-modelling approach with GPT-2 enables us to leverage recent advancements in language models [24], while alleviating challenges such as high inference computation costs and error accumulation that appeared in previous audio autoregressive models [8], [25]. The improvement is largely attributed to the shorter length of the LOA sequence. The continuous nature of LOA also potentially provides a richer representation power than the discrete tokens used in previous models [8], [13], [26].

Our experimental results demonstrate that *AudioLDM 2* achieves competitive performance on text-to-audio (TTA), and text-to-music (TTM) generation tasks, when evaluated on AudioCaps [27] and MusicCaps [8], respectively. On text-to-speech (TTS) generation tasks, *AudioLDM 2* achieves performance comparable with the SoTA by significantly outperforming a strong baseline FastSpeech2 [11]. In comparison to the original AudioLDM [4], *AudioLDM 2* contains a latent diffusion model that can be pretrained in a self-supervised manner, and enjoy the benefit of auto-regressive modeling of LOA with GPT-2 model. Besides, while retaining the same ability, *AudioLDM 2* shows substantial advancements over AudioLDM in quality, versatility, and capacity to generate speech with intelligible content. Overall, our contributions are as follows:

- We propose a novel and versatile audio generation model that is capable of performing conditional generation of audio, music, and intelligible speech.
- The proposed method is based on a universal representation of audio, which enables large-scale self-supervised pretraining of the core latent diffusion model without audio annotation and helps to combine the advantages of both the auto-regressive and the latent diffusion model.
- Our experiments shows three variants of *AudioLDM 2* achieves performance that match current state-of-the-art (SoTA) in text-to-audio, text-to-music, and text-to-speech generation on AudioCaps [27], MusicCaps [8], and LJSpeech [28] evaluation set, respectively.

II. RELATED WORK

A. Conditional Audio Generation

Audio generation is an emerging topic that focuses on modelling the generation of general audio, including recent models such as AudioGen [3], AudioLDM [4], and Make-an-Audio [15]. AudioGen treats audio generation as a conditional language modelling task, while the other two works

approach this task by latent diffusion. Studies on image-to-audio and video-to-audio generation, such as Im2Wav [29] and SpecVQGAN [30], are also areas of interest to researchers. Additionally, there are audio generation approaches that do not rely on conditioning, such as AudioLM [26], which performs audio language modelling based on a neural codec. Even though audio generation usually includes the topic of speech generation, previous works on text-to-audio generation tend to generate unintelligible speech [3], [4], [14], [15].

The field of audio generation encompasses sub-domains such as text-to-speech (TTS) and text-to-music (TTM). The former focuses on generating speech signals from transcriptions, while the latter involves creating a music clip from a textual description. Cutting-edge TTS models like FastSpeech2 [11], GradTTS [31], and NaturalSpeech [2] have made significant strides, producing speech of such high quality that it is nearly indistinguishable from human speech. Various techniques have been introduced to address speech generation in TTS, such as the monotonic alignment algorithm [32], which aligns phoneme features with spectrogram features, and a prosody predictor [11], used to guide model training and enhance expressiveness. Recent advances in TTM are evident in models like MusicLM [8], Noise2Music [33], MusicGen [34], and MeLoDy [13]. Similar to AudioLDM, the MusicLM model aligns music and language embeddings through contrastive pretraining modules, which enables text-free model optimization, alleviating the scarcity of music-text pairs. MusicLM also includes a semantic modeling stage based on w2v-BERT [35] to enhance the model performance. MusicGen uses a language modeling approach for music generation, enhanced with a mechanism for conditioning the model with melodic features for improved controllability. Meanwhile, MeLoDy, a diffusion model guided by language modeling, achieves significant computational reduction in music generation compared to MusicLM.

In this paper, we propose a unified framework for audio generation, which encompasses a breadth of topics including, but not limited to, speech, sound effects, and music generation.

B. Diffusion Models

Diffusion models [36], [37] have demonstrated high sample quality in a variety of tasks including image generation [38]–[40], image restoration [41], speech generation [42]–[44], and video generation [45], [46]. In the realm of speech or audio synthesis, these models have been explored for both mel-spectrogram generation [31], [47] and waveform generation [48]–[50]. However, the iterative nature of generation in a high-dimensional data space often results in slow training and inference speeds. One solution involves the use of diffusion models in a more restricted latent space, a strategy exemplified in image generation [23]. This idea has been adopted in various audio generation works, including AudioLDM [4], Make-An-Audio [15], and TANGO [51]. These works utilize latent diffusion models trained on a continuous latent space. On the other hand, there are also studies that explore diffusion in the discrete latent space. For instance, DiffSound [14] employs a discrete autoencoder to mitigate redundancy [52], [53] in the
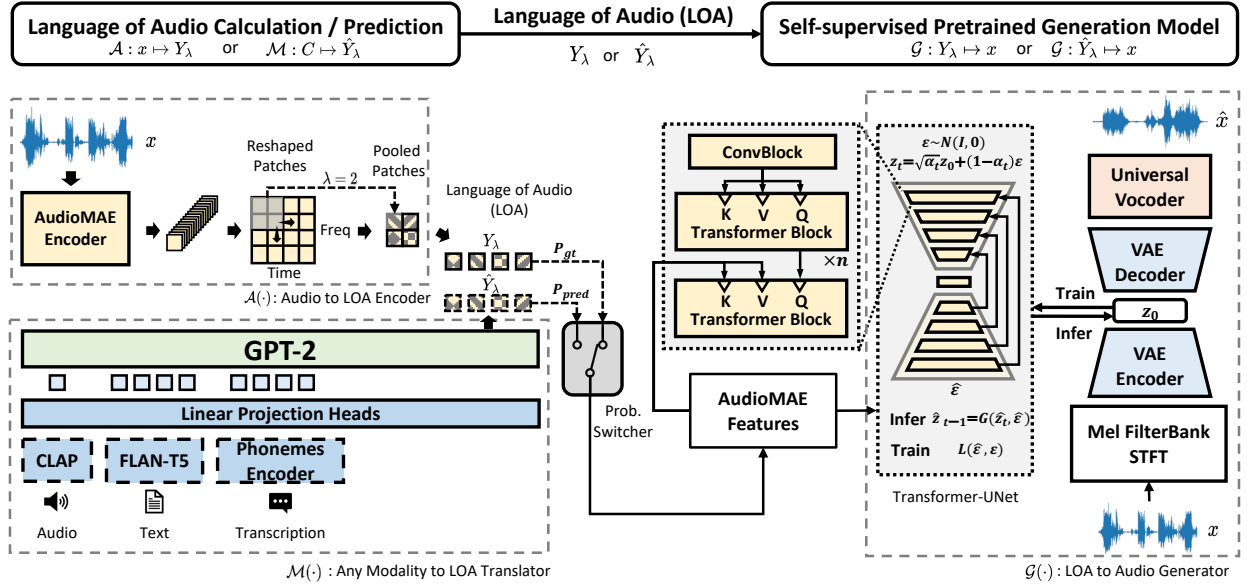
Fig. 1. The overview of the *AudioLDM 2* architecture. The AudioMAE feature is a proxy that bridges the *conditioning information to LOA translation* stage (modelled by GPT-2) and the *LOA to audio generation* stage (modelled by the latent diffusion model). The probabilistic switcher controls the probability of the latent diffusion model using the ground truth AudioMAE (P_{gt}) and the GPT-2 generated AudioMAE feature (P_{pred}) as the condition. Both the AudioMAE and latent diffusion models are self-supervised pre-trained with audio data.

audio waveform and create a compressed representation of mel-spectrograms. DiffSound utilizes text-conditional discrete diffusion models to generate discrete tokens.

III. AUDIOLDM 2

A. Overview

Let $x \in \mathbb{R}^{L_s}$ represent an audio signal, where L_s is the length of the audio samples in x. An audio generation process can be denoted as $\mathcal{H} : C \mapsto x$, where C is the conditioning information and $\mathcal{H}$ is the conditional audio generation system.

The direct generation of x from C is usually challenging [54]. Motivated by regeneration learning [55], we propose to utilize an intermediate feature Y, as an abstraction of x, to bridge the gap between C and x, as introduced in Section III-B1. We call the feature Y the language of audio (LOA). The LOA feature is calculated by $Y = \mathcal{A}(x)$ in which $\mathcal{A}$ performs audio to LOA encoding with a self-supervised representation learning module such as AudioMAE [21], [55]. The LOA feature should be a representation that is potentially easier to model compared with x and contain meaningful semantic information about x. As illustrated in Figure 1, with the intermediate representation Y, the overall audio generation process can be denoted as

$$\mathcal{H}_0 = \mathcal{G} \circ \mathcal{M} : C \mapsto \hat{Y} \mapsto x, \qquad (1)$$

where $\hat{Y}$ is the estimation of the ground truth LOA. As denoted in (1), the audio generation process of *AudioLDM 2* includes the following two steps:

(i) Conditioning information to LOA translation: The function $\mathcal{M} : C \mapsto \hat{Y}$ aims to produce the LOA Y based on C, which could be the conditional information from other modalities, such as audio and text. As a potentially better representation of C in terms of audio generation, the generated

$\hat{Y}$ will be used in later stages as the conditioning information for audio generation. We implement the function $\mathcal{M}$ with auto-regressive modelling, which is introduced in Section III-C.

(ii) LOA to audio generation: Followed by $\mathcal{M}$, function $\mathcal{G}$ accepts an LOA estimation $\hat{Y}$ as input condition and estimates the audio data x. During the training process, when the training data x is available, the ground truth Y will be also available using $\mathcal{A}(\cdot)$, allowing the optimization of $\mathcal{G}$ in a self-supervised manner. Specifically, instead of using the LOA estimation $\hat{Y} = \mathcal{M}(C)$, we condition the generation of x based on the $Y = \mathcal{A}(x)$, which can be formulated as

$$\mathcal{H}_1 = \mathcal{G} \circ \mathcal{A} : x \mapsto Y \mapsto \hat{x}. \qquad (2)$$

We introduce the detail of $\mathcal{A}(\cdot)$ in Section III-B. Since the process $\mathcal{H}_1$ only involves x as the training data, Equation (2) means model $\mathcal{G}$ can be optimized in a self-supervised manner without any audio annotation. This self-supervised scheme can alleviate the scarcity of the audio data labels [4] and provide a robust backbone for the overall generation system. Note that the self-supervised learning here does not refer to the entire *AudioLDM 2*, for example, the function $\mathcal{M}$ still needs paired data to optimize. We implement the function $\mathcal{G}$ with the latent diffusion model, which is introduced in Section III-D.

The following sections provide a detailed introduction to *AudioLDM 2*. In Section III-B, we discuss the audio representations employed in *AudioLDM 2*, including the AudioMAE and VAE features. These features also serve as the generation targets for the two stages within *AudioLDM 2*. Section III-C introduces the auto-regressive modeling of the AudioMAE feature with GPT-2. In Section III-D, we elucidate the process of generating audio waveforms via the latent diffusion model, which applies a VAE for feature compression and generates audio conditioned on the LOA. The LOA here can be based

on either ground truth or GPT-2-generated data, which corresponds to self-supervised training and joint training with GPT-2 (Section III-D3), respectively.

B. Audio Representation Learning

Motivated by MusicLM [8] and AudioLM [26], which perform semantic and acoustic modelling on two types of discrete representations [25], [56], we adopt a similar two-stage modelling approach. However, our work differs in that we work on the continuous semantic and acoustic representation, which can potentially provide richer information compared with discrete representations used by previous studies [25], [35]. In our work, we adopt AudioMAE [21] and variational autoencoder (VAE) [57] as the semantic and acoustic representation learning modules, respectively. Despite serving similar purposes, AudioMAE and VAE differ in architecture and objectives, yielding distinct representations. Further details on representation learning are provided below.

1) Semantic Representation Learning with the AudioMAE: To accurately represent diverse types of audio, encompassing speech, music, and sound effects, the LOA Y should effectively capture both the semantic and the acoustic details of audio signals. Therefore, we propose to use a self-supervised pretrained AudioMAE [21] as the representation extraction module for function $\mathcal{A}$ for its generality and high accuracy on the downstream audio classification task [21].

The audio masked autoencoder (AudioMAE) is an audio self-supervised pre-training model, which learns representations from unlabeled audio data without relying on manually labeled annotations. An AudioMAE consists of an encoder and a decoder, both realized with an architecture similar to the vision transformer (ViT) [58]. During self-supervised pretraining, input patches to the encoder, which are usually mel spectrograms, are randomly masked and the decoder learns to reconstruct the masked patches [21]. Compared with other audio self-supervised pretraining models, AudioMAE has the following two advantages:

(i) The AudioMAE has been verified to work well in the general audio domain. For example, an AudioMAE can be effectively pre-trained on AudioSet [59], with state-of-the-art performance on the downstream audio classification tasks. In comparison, typical audio self-supervised models focus on a specific domain, such as the MERT [60] on music and the HuBERT [61] on speech.

(ii) AudioMAE features are potentially better for generative tasks than other discriminative pre-training methods. Building upon the contrastive loss or next token prediction classification loss as the learning objective, previous systems such as wav2vec [62] and BYOL-A [63] utilize a discriminative approach during pre-training. In comparison, AudioMAE focuses on a generative process by learning the reconstruction of the masked patches.

For an input audio signal x, AudioMAE first calculates the log mel spectrogram $X \in \mathbb{R}^{T \times F}$, where T represents the time steps of the mel spectrogram, and F denotes the mel bins. The mel spectrogram X is then treated as an image and split into patches each of size $P \times P$, serving as the input
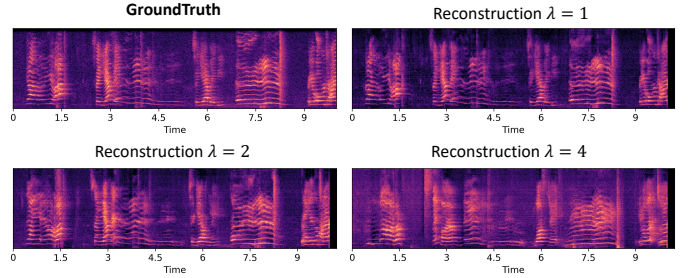

Fig. 2. The influence of λ on audio reconstruction from LOA Y_λ with the latent diffusion model. The reconstruction closely resembles the ground truth when $\lambda = 1$, suggesting that $Y_{\lambda=1}$ retains sufficient audio details. However, with $\lambda = 2$ or 4, the reconstruction diverges slightly from the original audio, indicating that while the post-processed AudioMAE feature may not include all details, it nonetheless accurately preserves semantic content.

for the AudioMAE encoder. The patch size P is typically designed to be a common factor of T and F. Patch splitting and embedding are performed using a convolutional neural network with a kernel size of P, a stride of P, and D output channels. This yields an output shape of $T' \times F' \times D$, where D is the AudioMAE embedding dimension, $T' = T/P$, and $F' = F/P$. The resulting output feature of the AudioMAE encoder, $E \in \mathbb{R}^{T' \times F' \times D}$, has the same shape as the input and is usually treated as the feature for downstream tasks after pretraining [21].

2) AudioMAE Feature Post Processing: As shown in Figure 1, once the AudioMAE features E are computed, we introduce an additional pooling step to aggregate E into Y_λ, where $\lambda \in I^+$ represents a hyper-parameter used in the post-processing pooling step. This pooling step aims to reduce the sequence length, facilitating easier estimation in the function $\mathcal{M}$. Specifically, we perform a two-dimensional average-max pooling [52] on the first two dimensions of $E \in \mathbb{R}^{T' \times F' \times D}$, in which the pooling kernel size and stride have the same value $\lambda \in I^+$. The two-dimensional pooling operation can help to preserve the time-frequency relationship in the output. The final output after pooling, Y_λ, is reshaped into a embedding sequence with shape $L_\lambda \times D$, in which $L_\lambda = T'F'/\lambda^2$. To facilitate implementation, λ is chosen so that L_λ is always a positive integer. We demonstrate the effect of different choices of λ in Figure 2. In the remaining sections of this paper, if λ is not specified, we'll refer to Y_λ simply as Y.

3) Acoustic Representation Learning with VAE: We use a VAE for feature compression and for learning an audio representation z, which has a significantly smaller dimension than x [4]. The VAE we used in this work is a convolutional architecture that consists of encoders with down-sampling and decoders with up-sampling following the architecture described in [4]. The forward pass of the VAE can be formulated as $\mathcal{V} : X \mapsto z \mapsto \hat{X}$, where X is the mel-spectrogram of x and $\hat{X}$ is the reconstruction of x. The reconstruction $\hat{X}$ can be converted to the audio waveform $\hat{x}$ using a pretrained HiFiGAN vocoder [64]. Following AudioLDM [4], we calculate a reconstruction loss and a discriminative loss based on X and $\hat{X}$ to optimize the parameters of the VAE. We also calculate the KL divergence between z and a standard Gaussian ($\mu = 0$, $\sigma^2 = 1$) as a loss function to limit the

VAE latent space visualized

- ■ Brushing teeth
- ■ Clapping
- ■ Engine
- ■ Hand saw
- ■ Pouring water
- ■ Sea waves
- ■ Sheep
- ■ Snoring
- ■ Toilet flush
- ■ Train

AudioMAE latent space visualized

- ■ Brushing teeth
- ■ Clapping
- ■ Engine
- ■ Hand saw
- ■ Pouring water
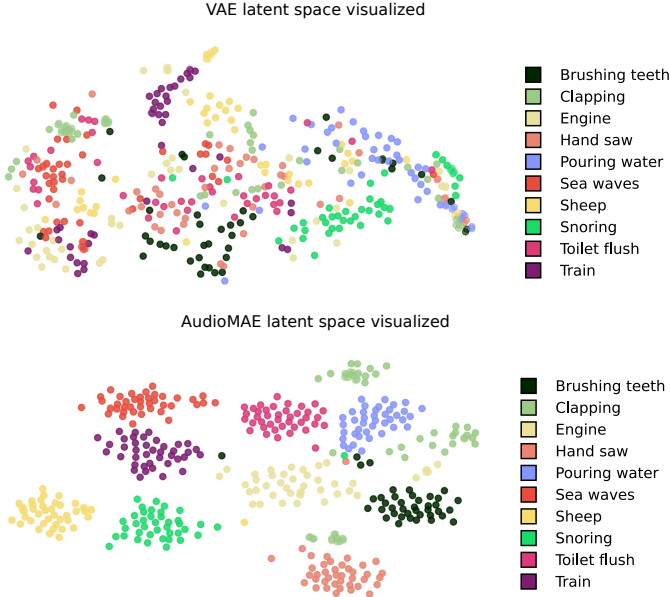- ■ Sea waves
- ■ Sheep
- ■ Snoring
- ■ Toilet flush
- ■ Train

Fig. 3. Visualization of the latent space based on tSNE and ten randomly selected classes in the ESC50 [65] dataset. Each point in the figure represents an audio clip. The AudioMAE feature space tends to group similar audio clips together, indicating more semantic structure than in the VAE feature.

variance of the VAE latent space.

4) Comparison between AudioMAE and VAE: Since both AudioMAE and VAE are based on autoencoders for representation learning, one might wonder why we use a VAE for representation learning instead of directly modeling the AudioMAE latent space. Part of the reason is that AudioMAE does not primarily focus on reconstruction quality, and its latent space compression ratio is not as high as that of the VAE. On the other hand, the VAE exhibits good reconstruction ability and a higher compression level than AudioMAE, making VAE more suitable for mel-spectrogram compression. Furthermore, as shown in Figure 3, we visualize the latent representation of AudioMAE and VAE on the ESC-50 [65] dataset using tSNE [66]. The visualization demonstrates that the latent representation of AudioMAE can group similar audio at a closer region in the latent space. In contrast, the representation of VAE exhibits more overlap between different audio classes. This indicates that the representations for the AudioMAE and VAE are distinct. AudioMAE contains more information on the semantic side, while VAE representation is less semantically structured. Therefore according to the definition of LOA in Section III-A, AudioMAE is more suitable than VAE on calculating the LOA.

C. Conditioning Information to LOA Translation with GPT-2

This subsection introduces the design of the function $\mathcal{M}$. As introduced in Section III-A, the input to the model $\mathcal{G} : Y \mapsto x$ can be calculated using the AudioMAE. However, during inference, when we perform audio generation with the condition C, the ground truth LOA $Y = \mathcal{A}(x)$ is unavailable. Therefore, we need another model that can generate $\hat{Y}$ given C, denoted by $\mathcal{M}_\theta : C \to \hat{Y}$, where θ represents trainable parameters.

Specifically, we treat the generation of Y as a language modelling task and choose the GPT-2 (Generative Pre-trained Transformer 2) [22] model as the backbone. GPT-2 is based on a transformer architecture and was originally trained on 8 million documents for a total of 40 GB of text using an unsupervised learning approach [22]. GPT-2 has been used in a variety of natural language processing tasks, such as text completion, question answering, and language translation [67], [68]. Initialized with pre-trained weights, we finetune the GPT-2 model based on teacher forcing [69], so that during model training, $\hat{y}_l$ will be generated based on both the condition C and the ground truth sequence $y_1, ..., y_{l-1}$, where y_l is the $l-$th vector in LOA sequence Y. Specifically, the GPT-2 model $\mathcal{M}_\theta$ is trained to maximize the likelihood of a sequence $Pr(y_1, y_2, ..., y_L|C)$, which can be interpreted into the following optimization objective:

$$\text{argmax}_\theta \ \mathbb{E}_C \ [Pr(y_1|C;\theta) \prod_{l=2}^{L} Pr(y_l|y_1, ..., y_{l-1}, C; \theta)], \quad (3)$$

where $\mathbb{E}_C$ represents the expectation operator with respect to the variable C. We calculate the mean squared error loss [54] between y_l and $\hat{y}_l = \mathcal{M}_\theta(y_1, ..., y_{l-1}, C)$ to optimize Equation (3). We directly optimize the regression of continuous vectors y_l, without discretizing the AudioMAE feature space and estimating the token index. The condition C in Equation (3) can encompass a flexible range of data representations, including audio representations, text embeddings, phoneme embeddings, or visual clues. We adopt the mixture of experts [70] approach and use multiple encoders as feature extractors to calculate C. Given K systems as the feature extraction modules, the shape of the output from the k-th system $C_k, k \in \{1, ..., K\}$ is $L_k \times D_k$, in which L_k is the sequence length of the k-th system and D_k is the dimension of the feature. We apply a linear transformation layer after the output of each feature extraction module to unify the embedding dimension to D_0 for an easier process of the GPT-2 model. For modules that extract global features from the input without sequential information, such as CLAP [71] or ImageBind [18], we have $L_k = 1$. The final condition $C = [C_1, ...C_K]$ is a concatenation of C_k along the sequence length dimension. The final condition C has a shape of $L \times D_0$, where $L = \sum_{k=1}^{K} L_k$. We introduce several condition modules we used in this paper as follows.

CLAP or contrastive language and audio pretraining [71], is a system that learns a joint audio-text embedding space, in which paired audio and language data have a closer distance in the latent space. CLAP has been successfully applied as a conditioning module to audio generation such as AudioLDM [4]. In this study, we employ a pre-trained CLAP[1] text encoder as the default conditioning module for extracting text embeddings as conditions. However, in scenarios where text captions (e.g., "A man is speaking happily with background static noise") are unavailable, such as for text-to-speech tasks, we use the CLAP audio encoder as the conditioning module instead of using CLAP text encoder, in the same way as [4].

---

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

FLAN-T5. The CLAP model, as a module that calculates global-level conditions, has been found to have issues in capturing the temporal information in the text data [72]. To allow for this, we use another pretrained text encoder to capture the semantic information of the textual input, which might contain useful details such as temporal orders. Specifically, we utilize FLAN-T5 [73], which is an enhanced version of the text-to-text transfer transformer (T5) model [74] based on the finetuning on a mixture of tasks[2].

Phoneme Encoder is a widely adopted module in text-to-speech research for extracting helpful information regarding phonemes [2], [11], which are the smallest units of sound in a language that can distinguish one word from another [75]. In this work, we follow the structure introduced in NaturalSpeech [2] to build a phoneme encoder, in the form of a stack of transformer encoder layers. We preprocess the textual input into phonemes using the open-source tool Espeak phonemizers [3] and append a stop token after each phoneme sequence to mark the end of the sequence for the transformer model.

Except for the phoneme encoder, which does not have a readily available pre-trained weights, the parameters of all other pre-trained feature extraction models are kept frozen during the experiment.

D. LOA to Audio Generation with Latent Diffusion Model

We model the process $\mathcal{G} : Y \mapsto x$ with a latent diffusion model (LDM) [23], which is a variant of the denoising diffusion probabilistic models (DDPM) [36]. In contrast to DDPM, which directly models the training data, the LDM learns the reverse diffusion process in a variational autoencoder (VAE)-based compressed latent space [76], which can reduce the computational cost. Similar ideas have been adapted to audio generation, such as AudioLDM [4].

1) Latent Diffusion Model: We follow the formulation in [36] to implement the LDM. Given a VAE representation z, the forward transition is defined as a T steps Markov process in a way that does not include trainable parameters. Given the data z_{t-1} at diffusion step $t-1$, the data distribution of z_t at step $t \in 2,...,T$ can be formulated as

$$q(z_t|z_{t-1}) = \sqrt{1-\beta_t}z_{t-1} + \sqrt{\beta_t}\epsilon_t, \quad (4)$$

in which the noise schedule hyper-parameter $\beta_t \in [0,1]$ determines how quickly the noise is blended into the data. By recursive substitution of $q(z_t|z_{t-1})$ in Equation (4) [36], we can derive the distribution of z_t given z_0 as

$$q(z_t|z_0) = \sqrt{\alpha_t}z_0 + \sqrt{1-\alpha_t}\epsilon_t, \quad (5)$$

where $\alpha_t = \prod_{t=1}^t 1 - \beta_t$ and $\epsilon_t \sim N(0,I)$. At the final step $t = T$, the distribution of z_t will be close to a standard Gaussian distribution [36].

The LDM learns a backward transition from the prior distribution $N(0,I)$ to the data distribution z. The reverse process models the conditional distribution $Pr(z_{0...T}|Y;\phi) =$

$Pr(z_0|z_1,Y;\phi)\prod_{t=2}^T Pr(z_{t-1}|z_t,Y;\phi) \cdot Pr(z_T)$, in which Y is the LOA as the condition signal and the ϕ denotes the parameter of the model for learning the reverse diffusion. If we marginalize $z_{1...T}$ we can derive the lower bound of $\log[Pr(z_0|Y;\phi)]$ based on the evidence lower bound (ELBO) and Bayes' rule [36]:

$$\log[Pr(z_0|Y;\phi)] \geq \log[Pr(z_0|z_1,Y;\phi)]-$$
$$\sum_{t=2}^T KL[Pr(z_{t-1}|z_t,Y;\phi)||q(z_{t-1}|z_t,z_0)], \quad (6)$$

where $KL(\cdot)$ is the function for calculating KL divergence, and $q(z_{t-1}|z_t,z_0))$ is the target conditional diffusion distribution that has a closed-form solution given z_0 and z_t [36]. Following [36], we can derive the loss function that maximizes the lower bound of Equation (6) as:

$$\text{argmin}_\phi[\mathbb{E}_{z_0, Y, t\sim\{1,...,T\}}||\mathcal{G}(\sqrt{\alpha_t}z_0+\sqrt{1-\alpha_t}\epsilon_t,t,Y;\phi)-\epsilon_t||].$$
$$(7)$$

As shown in Figure 1, we utilize a Transformer-UNet (T-UNet) architecture as the function $\mathcal{G}$ in Equation (7), which is similar to the UNet used in AudioLDM [4] but with more transformer layers. The T-UNet architecture consists of a series of encoders with downsampling and a series of decoders with upsampling, and there are skip connections between encoders and decoders at the same scale. To enhance the modelling capacity of the T-UNet, we insert multiple transformer blocks after the convolution operation in each encoder and decoder block. Specifically, we have $n_{\text{trans}} + 1$ transformer blocks, in which the first n_{trans} transformer blocks are a stack of self-attention layers [77] and feed-forward networks. To incorporate the condition information Y from the ground truth LOA or $\hat{Y}$ from $\mathcal{M}(\cdot)$ (Section III-C), as shown in Figure 1, the last transformer block changes the self-attention layer to cross-attention, which accepts the LOA as key and value and fuses with the feature from the previous transformer block as the query. Except for text-to-speech generation, we add an extra cross-attention layer in the transformer block to accept the text embedding from FLAN-T5 [73] as an extra condition to enhance the audio-text relationship learning.
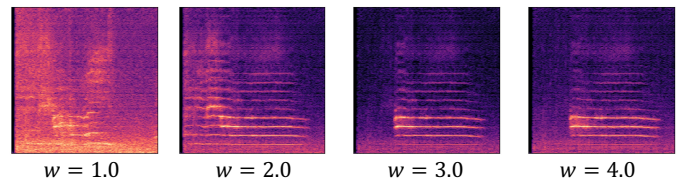

Fig. 4. The samples generated with different classifier-free guidance scales. The text prompt is *"A cat is meowing"*.

2) Classifier-free Guidance: For diffusion models, controllable generation can be achieved by introducing guidance at each sampling step. Classifier-free guidance [78], [79] (CFG) has been the state-of-the-art technique for guiding diffusion models. During training, we randomly discard our condition Y in Equation (7) with a fixed probability (e.g., 10%) to train both the conditional LDMs $\mathcal{G}(z_t,t,Y;\phi)$ and

the unconditional LDMs $\mathcal{G}(z_t, t, \phi)$. For generation, we use LOA $\hat{Y}$ or Y as the condition and perform sampling with a modified noise estimation $\mathcal{G}'(z_t, t, Y; \phi)$:

$$\mathcal{G}'(z_t, t, Y; \phi) = w\mathcal{G}(z_t, t; \phi) + (1 - w)\mathcal{G}(z_t, t, Y; \phi), \quad (8)$$

where w determines the guidance scale.

3) Joint Finetuning: We perform joint finetuning with the GPT-2 and latent diffusion models based on Equation (1), (7), and (3). As demonstrated by Table V, we found that joint finetuning significantly enhances the overall performance of the *AudioLDM 2* system. As depicted in Figure 1, the *probabilistic switcher* controls the source of the conditioning signal during the joint training process. During training, the switcher dynamically chooses between ground truth AudioMAE features and GPT-generated AudioMAE features, with probabilities set to P_{gt} and P_{pred}, respectively.

IV. EXPERIMENT SETUP

A. Dataset

The datasets used in this work include AudioSet (AS) [59], WavCaps [80], AudioCaps (AC) [27], VGGSound (VS) [81], Free Music Archive (FMA) [82], Million Song Dataset (MSD) [83], LJSpeech (LJS) [28], and GigaSpeech (GGS) [84]. AudioSet is the largest audio classification dataset at the time of writing, with around two million ten-seconds of audio and 527 different classes. WavCaps is a dataset with ChatGPT-assisted weakly-labeled audio captions. WavCaps contains 403,050 audio clips with an average duration of 68 seconds. AudioCaps is a subset of AudioSet with handcrafted captions, containing about 46,000 ten-second audio clips. VGGSound is a large-scale single-label audio-visual dataset, which contains over 200,000 videos. We only utilize the audio data and the labels in the VGGSound. FMA is a large music dataset without captions, containing 106,574 music tracks from 16,341 artists and 14,854 albums. For the Million Song Dataset, we only utilize the labelled subset proposed in [85], which contains around 510,000 music tracks with metadata such as tags, titles, and artist names. LJSpeech is a single-speaker speech dataset with 13,100 short audio clips and detailed transcriptions. GigaSpeech is a multi-speaker large-scale English speech recognition corpus with around 10,000 hours of audio labeled with transcriptions. The test and development set of GigaSpeech are not included during training. All the audio data used in this work are resampled to 16 kHz for easier comparison with previous works [4], [15]. We use only the audio data with paired text labels to train the GPT-2 model by optimizing Equation (3). We train the latent diffusion model with all the audio data regardless of annotation by optimizing the objective in Equation (6) in a self-supervised manner.

B. Evaluation Metrics

We mainly focus on the text-to-audio generation task to evaluate the effectiveness of *AudioLDM 2*. We follow the evaluation protocol of AudioGen [3], which calculates both objective metrics such as Frechet Audio Distance (FAD), Kullback-Leibler Divergence (KL), and subjective metrics including Overall Impression (OVL) and Audio and Text Relation (REL). We also include an additional metric CLAP score [15] to measure the correspondancy between the generated audio and text prompt. FAD is a reference-free audio quality measure that is calculated based on the distribution distance between the feature of the target and generated audios, extracted from the VGGish [86] model. KL divergence measures the similarity between the generated and target audio with the label calculated by the audio tagging model, Patch-out Transformer [87], in the same way as AudioGen [3]. CLAP score measures the similarity between audio and text based on a pair of pretrained audio and text encoders [71], given by

$$\text{CLAPScore}(x, r) = \frac{\vec{e}_x \cdot \vec{e}_r}{\max(\|\vec{e}_x\|\|\vec{e}_r\|, \epsilon)}, \quad (9)$$

where x and r denote audio and text data, respectively, ϵ is a small value that can avoid zero division, $\vec{e}_a$ is the output of the CLAP audio encoder and $\vec{e}_r$ is the output of CLAP text encoder. The value range of the CLAP score is between -1 and 1 and a larger value indicates a stronger correlation between audio and text information.

We use a similar evaluation protocol for text-to-music generation. For the text-to-speech task, we utilize the commonly used mean opinion score (MOS) for evaluation [75].

C. Subjective Evaluation

We use Amazon Mechanical Turk[4], a crowd-sourced platform, to evaluate subjective metrics including OVL, REL, and MOS. The instructions on how to perform evaluation are clearly illustrated for the raters with examples. Specifically, for OVL, raters were asked *How would you rate the overall quality of this music? Consider its resemblance to real-world audio and its naturalness,* with a five-point scale ranging from *5-Excellent quality* to *1-Bad quality*. Similarly, for REL, the question posed was, *How would you rate the relevance of music to the text description?* with a similar five-point scale for responses. In evaluating MOS, the question was, *How natural does this recording sound? Take into account emotion, prosody, and other human-like details,* with options ranging from *completely unnatural speech* to *perfectly natural speech*. To ensure the credibility of the evaluation result, we set requirements for the crowd-source worker with a minimum average approval rate of 60% and with at least 50 approvals in the record. Each audio clip is evaluated by at least 10 different raters. All three subjective metrics have a Likert scale [88] between one and five, where a larger number indicates better performance. Study raters received payment at or above the US minimum wage. We average the scores among all raters and samples as the final score for a system.

D. Model Architecture Details

We perform the experiment with two sizes of the latent diffusion model, *AudioLDM 2* and *AudioLDM 2-Large*, with transformer layer numbers $n_{trans} = 2$ and $n_{trans} = 6$ (Section III-D), respectively. We use a pre-trained AudioMAE[5]

---

[4]https://requester.mturk.com/
[5]https://github.com/facebookresearch/AudioMAE

with a patch size of 16×16 and no overlapping, resulting in a 768-dimension feature sequence with length 512 for every ten seconds of mel spectrogram. In a similar way to the idea introduced in [89], on calculating the LOA Y, we gather the output of the last 16 transformer layers from the AudioMAE encoder and perform averaging as the final Y. We perform self-supervised pre-training on both *AudioLDM 2* and *AudioLDM 2-Large* with the audio data mentioned in Section IV-A. The GPT-2 model we employ has an embedding dimension of 768 with 12 layers of transformers. For joint fine-tuning, we set the probability of using ground truth LOA Y and LOA estimation $\hat{Y}$ as $P_{\text{gt}} = 0.25$, and $P_{\text{pred}} = 0.75$, respectively.

TABLE I
THE SETUP OF THE PRIMARY EXPERIMENTS WE PERFORMED. FULL REPRESENTS A COMBINATION OF FIVE DIFFERENT DATASETS, INCLUDING AC, AS, WC, VS, AND MSD. THE MODEL WITH † USE CLAP TEXT ENCODER AND FLAN-T5 TO CALCULATE CONDITIONS WHILE THE MODEL WITH ‡ USES CLAP AUDIO ENCODER AND THE PHONEME ENCODER AS THE CONDITIONAL MODULES.

Model	λ	Param	Dataset	Task
AudioLDM 2-AC†	8	346M	AC	TTA
AudioLDM 2-MSD†	8	346M	MSD	TTM
AudioLDM 2-Full†	8	346M	FULL	TTA/TTM
AudioLDM 2-AC-Large†	8	712M	AC	TTA
AudioLDM 2-Full-Large†	8	712M	FULL	TTA/TTM
AudioLDM 2-LJS‡	1	346M	LJS	TTS
AudioLDM 2-LJS-Pretrained‡	1	346M	LJS+GGS	TTS

Table I summarize the experiments we performed in this paper. For the generation of audio and music, we combine the text embeddings from the CLAP text encoder and FLAN-T5 as conditioning and designate $Y_{\lambda=8}$ as the target sequence for GPT. The conditioning modules for speech generation are configured differently, primarily due to the need to better preserve the fine-grained phoneme information in speech signals through a smaller λ value. Thus, for speech generation, we concatenate the output of CLAP audio encoder and the phoneme encoder as the input sequence of the GPT-2 model, and designate $Y_{\lambda=1}$ as the target sequence to retain more details. For the speech data, since there are no available audio captions (different from transcriptions), we adopt a similar approach as AudioLDM [4] to utilize the CLAP audio encoder to compute the embedding as a condition during model training, and employ the CLAP text encoder during inference. This method also facilitates prompt-based speaker control, as demonstrated in Figure 6.

E. Training and Inference Setup

The latent diffusion model and the GPT-2 model are initially trained separately. We randomly choose $\lambda \in \{1, 2, 4, 8\}$ during pre-training of the latent diffusion model to enhance the model robustness under conditions Y_λ with different λ. Y_λ is only used as key and value in the T-UNet cross-attention layers therefore Y_λ can have varying length. We train the latent diffusion model based on 10 seconds of random segment from the training set. For easier modeling of the T-UNet, we zero-pad the 10 seconds of audio segment into 10.24 seconds

during model training. We train the latent diffusion model and finetune the GPT-2 model on eight NVIDIA A100 80GB GPUs. We follow the settings described in AudioLDM [4] and change the default classifier-free guidance scale during the Denoising Diffusion Implicit Models (DDIM) [90] sampling to 3.5. For both GPT-2 finetuning and the latent diffusion model, we utilize the AdamW [91] optimizer with a learning rate of 10^{-4} and 10000 steps of linear warming up without decay.

V. RESULT

We evaluated our proposed system on three primary audio generation tasks: text-to-audio, text-to-music, and text-to-speech. The three basic systems were trained on three different datasets: AudioCaps (general audio), MSD (music), and LJSpeech (speech), and are denoted as *AudioLDM 2-AC*, *AudioLDM 2-MSD*, and *AudioLDM 2-LJS*, respectively. The model *AudioLDM 2-Full* represents a version capable of performing both audio and music generation simultaneously, with training data scaled up to 29510 hours, including all available data mentioned in Section IV-A. In contrast with AudioLDM [4], we do not perform additional model finetuning on AudioCaps for model trained with the full-scale datasets. Models with the suffix *Large* indicate larger-sized model variants, such as *AudioLDM 2-Full-Large*.

A. Text-to-Audio Generation

We compare the performance of our proposed model with several state-of-the-art systems, including AudioGen-Large [3], Make-an-Audio [15], AudioLDM [4], Make-an-Audio 2 [92], and TANGO [51]. To generate the samples for subjective evaluation, we adopt AudioLDM-M, an AudioLDM with 652M parameters, from HuggingFace[6] and run with 100 reverse diffusion steps. The result of Make-an-Audio 2 is provided by the author [92]. We use the pre-trained TANGO model open-sourced on GitHub[7] to reproduce their result.

As shown in Table II, our proposed *AudioLDM 2-AC* significantly outperforms the previous systems across all three objective metrics. The previous best-performing system, TANGO, achieves a CLAP score of 17.6, while our proposed system surpasses it with a substantially higher CLAP score of 24.9. *AudioLDM 2-Large* also attains the best KL divergence score of 0.98, considerably improving upon the previous SoTA of 1.27. For the FAD score, our model reaches 1.42, establishing a new SoTA for text-to-audio generation. Our subjective evaluation results are mostly consistent with the objective metrics, confirming the effectiveness of *AudioLDM 2-AC*, which achieves an OVL of 3.88 and a REL of 3.90, surpassing AudioLDM and the previous SoTA TANGO by a significant margin. The difference between *AudioLDM 2-AC* and the GroundTruth, which are real audios from the AudioCaps dataset [27], is merely 0.16 and 0.18 for OVL and REL, respectively, demonstrating the strong performance of our proposed system. The AudioLDM-M we used is not finetuned on the AudioCaps dataset, which may explain its degraded performance compared with the metric score reported

---

[6]https://huggingface.co/spaces/haoheliu/audioldm-text-to-audio-generation
[7]https://github.com/declare-lab/tango

TABLE II

COMPARISON OF MODEL PERFORMANCES ON THE AUDIOCAPS EVALUATION SET. GT-AUDIOMAE DENOTE DIRECTLY APPLYING THE GROUND TRUTH "LANGUAGE OF AUDIO" Y, TO THE FUNCTION $\mathcal{G}$ FOR AUDIO GENERATION, AS DETAILED IN SECTION III-A. *AudioLDM 2* SIGNIFICANTLY SURPASSES PREVIOUS METHODS IN BOTH SUBJECTIVE AND OBJECTIVE ASSESSMENTS. ALL MODELS ARE TRAINED USING THE AUDIOCAPS TRAINING SUBSET. MODELS MARKED WITH ∗ ARE EXCLUSIVELY TRAINED ON THIS SUBSET, WHILE THOSE WITH # ARE FINE-TUNED ON IT.

Model	Duration (h)	Param	FAD↓	KL↓	CLAP↑	OVL ↑	REL ↑
GroundTruth	-	-	-	-	0.251	4.04	4.08
GT-AudioMAE	-	-	1.84	0.19	0.239	3.87	4.02
AudioGen-Large	6824	1 B	1.82	1.69	-	-	-
Make-an-Audio	3000	453 M	2.66	1.61	-	-	-
AudioLDM-Large#	9031	739 M	1.96	1.59	-	-	-
AudioLDM-M	9031	416 M	4.53	1.99	0.141	3.61	3.55
Make-an-Audio 2	3700	937 M	2.05	1.27	0.173	3.68	3.62
TANGO∗	145	866 M	1.73	1.27	0.176	3.75	3.72
AudioLDM 2-AC∗	145	346 M	1.67	1.01	**0.249**	3.88	**3.90**
AudioLDM 2-Full	29510	346 M	1.78	1.60	0.191	3.83	3.77
AudioLDM 2-AC-Large∗	145	712 M	**1.42**	**0.98**	0.243	**3.89**	3.87
AudioLDM 2-Full-Large	29510	712 M	1.86	1.64	0.182	3.79	3.80

A pencil scribbling on a notepad.

A kitten mewing for attention.

Magical fairies laughter echoing through an enchanted forest.

A monkey laughs before getting hit on the head by a large atomic bomb.

A accordion is speaking

A classical orchestra performing a grand symphony.

A playful glockenspiel in a whimsical children's song, sparking imaginations.

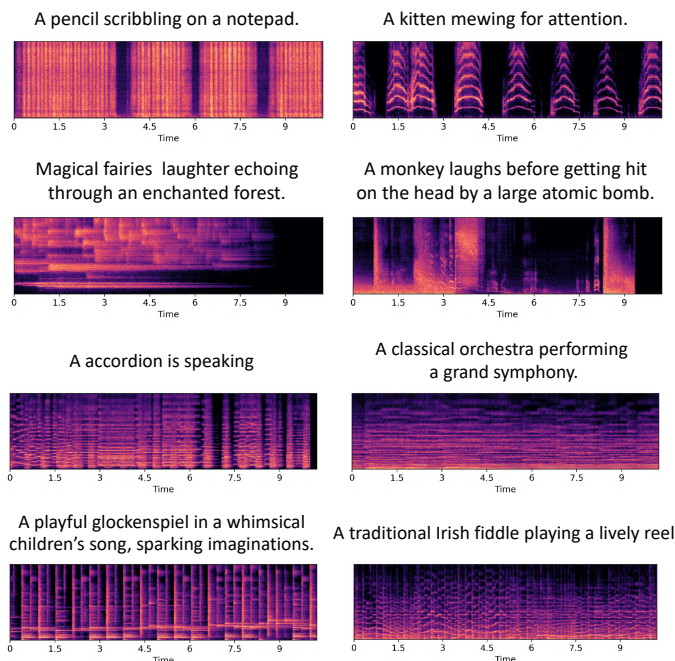A traditional Irish fiddle playing a lively reel

Fig. 5. Examples for text-to-audio generation.

in [4]. We observe the trend of overfitting during *AudioLDM 2* training on the AudioCaps training set, due to the limited dataset size. To address this issue we measure the FAD score on the AudioCaps validation set every five epochs and treat the checkpoint before the FAD result shows degradations as the final model.

To investigate the scalability of *AudioLDM 2-AC-Large* in terms of model size and dataset scale, we further trained *AudioLDM 2* on a much larger dataset containing 29,510 hours of data using two different model sizes. To avoid the overfitting issue and potentially misleading objective metrics result, the model trained with a larger dataset, including *AudioLDM 2-Full* and *AudioLDM 2-Full-Large* are not fine-tuned on the AudioCaps training set, as performed by previous

works [4]. As shown in Table II, the FAD score generally shows improvement after scaling up the model size, while the KL divergence and CLAP scores do not exhibit clear improvements, indicating that scaling the model size might be more beneficial for enhancing audio quality than audio-text relations. Despite the significant increase in training data, we did not observe significant improvements in the objective evaluation metrics. On the contrary, all three metrics showed degraded performance after training on more data. This is potentially because our test set has a limited distribution, while the large-scale training data covers a much wider distribution. The mismatch between the training and test data distributions results in poorer objective scores.

Nevertheless, when compared with the AudioLDM-M (FAD 4.53) in Table II, which is also a large-scale pre-trained text-to-audio model without finetuning on AudioCaps, *AudioLDM 2* with full-scale training data achieves significantly better performance (FAD 1.42 ∼2.13), showing a substantial improvement over AudioLDM-M.

B. Text-to-Music Generation

In this section, we compare our proposed model with other text-to-music generation models, including MusicGen [34], MusicLM [8], MeLoDy [13], Mousai [93], AudioLDM [4], and Riffusion [6]. The output of AudioLDM is obtained in the same way as Table II. MusicGen is reproduced using the official Github repository[8].

As shown in Table III, our proposed method significantly outperforms these strong baselines. For instance, *AudioLDM 2-Full* outperforms MusicGen by 36%, 11%, and 3.4% on FAD, KL and CLAP scores, respectively. The *AudioLDM 2-MSD* model, which is only trained on music data, does not achieve better performance on objective metrics than the more general *AudioLDM 2-Full*. This result suggests that learning audio generation from a general perspective can benefit the performance in specialised domains as well, demonstrating the advantages of our proposed general framework. The general

[8]https://github.com/facebookresearch/audiocraft

TABLE III
PERFORMANCE COMPARISON ON THE MUSICCAPS EVALUATION SET. THE SUPERSCRIPT ⋆ INDICATES RESULTS REPRODUCED USING PUBLICLY AVAILABLE IMPLEMENTATIONS. THE OPEN-SOURCE VERSION OF MUSICGEN-MEDIUM EXCLUDES VOCAL SOUNDS, RESULTING IN SLIGHTLY INFERIOR PERFORMANCE COMPARED TO THE ORIGINAL REPORT [34]. GT-AUDIOMAE DENOTE DIRECTLY APPLYING THE GROUND TRUTH "LANGUAGE OF AUDIO" Y, TO THE FUNCTION $\mathcal{G}$ FOR AUDIO GENERATION, AS DETAILED IN SECTION III-A. ALL GENERATED AUDIO CLIPS WERE RESAMPLED TO 16KHZ PRIOR TO EVALUATION.

Model	FAD↓	KL↓	CLAP↑	OVL↑	REL↑
GroundTruth	-	-	0.253	3.82	4.26
GT-AudioMAE	2.18	0.27	0.257	3.59	3.92
Riffusion	14.80	2.06	0.190	-	-
Mousai	7.50	1.59	-	-	-
MeLoDy	5.41	-	-	-	-
MusicLM	4.00	-	-	-	-
MusicGen-Medium	3.4	1.23	**0.320**	-	-
MusicGen-Medium⋆	4.89	1.35	0.291	3.37	3.38
AudioLDM-M⋆	3.20	1.29	0.360	3.03	3.25
AudioLDM 2-MSD	4.47	1.32	0.294	**3.41**	3.30
AudioLDM 2-Full	**3.13**	**1.20**	0.301	3.34	**3.54**

model *AudioLDM 2-Full* achieves a significantly higher 3.54 REL score than the other systems, indicating better textual understanding ability. The AudioLDM-M model achieves a significantly higher CLAP score than the remaining systems, which may stem from being directly conditioned by the same CLAP model during training. Therefore, the CLAP score value of AudioLDM-M in Table III is only provided for reference and may not reflect the true performance of the model, as also indicated by the subjective evaluation score. The high performance of AudioLDM may also stem from the diversity of audio training data, which also includes music and sound effects, which further supports the benefits of training a general-purpose model. However, the subjective evaluation in Table III indicates that the subjective performance of AudioLDM-M is not as good as suggested by the objective metrics. Since MeLoDy and MusicLM are not open-sourced, some of their objective and subjective metrics scores are not available for comparison. Due to the substantially lower objective scores, Riffusion and Mousai are not included in our subjective evaluation against other baseline models.

C. Text-to-Speech Generation

We compare our proposed model with the widely-adopted FastSpeech2[9] model on the LJSpeech test set. To study the upper bound of our system, we add a setting called GT-AudioMAE that utilizes the ground truth LOA Y to the function $\mathcal{G}$ for audio generations. Our proposed *AudioLDM 2-LJS* is trained on the LJSpeech training split. To further explore the potential of our system, we pre-train the GPT-2 model in function $\mathcal{M}$ on the GigaSpeech dataset before finetuning on LJSpeech. This version is denoted as *AudioLDM 2-LJS-Pretrained*.

As shown in Table IV, with the pre-trained GPT-2 model, *AudioLDM 2-LJS-Pretrained* achieves a MOS of 4.00, significantly outperforming FastSpeech2. Our subjective evaluation
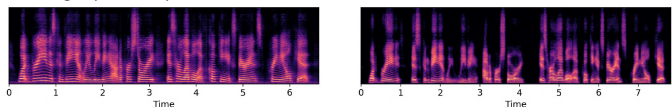
---

[9]https://huggingface.co/facebook/fastspeech2-en-ljspeech

TABLE IV
TEXT-TO-SPEECH PERFORMANCE EVALUATED ON THE LJSPEECH TEST SET.

Model	Mean Opinion Score↑
GroundTruth	4.63 ± 0.08
GT-AudioMAE	4.14 ± 0.13
FastSpeech2	3.78 ± 0.15
AudioLDM 2-LJS	3.65 ± 0.21
AudioLDM 2-LJS-Pretrained	4.00 ± 0.13

Text: I can heat things up.
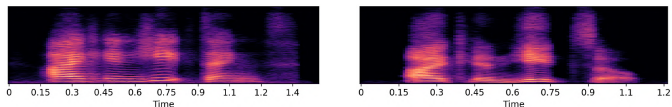
Fig. 6. Examples of speaker-prompted text-to-speech generation. We use speaker prompts to describe the characteristics of the speaker and provide the model with the text transcription.

shows *AudioLDM 2-LJS-Pretrained* exhibits greater fluctuations in emotion, punctuation, and tone. This demonstrates the benefits of pretraining on diverse datasets like GigaSpeech before finetuning on smaller corpora. Without pretraining, our proposed model still achieves a competitive MOS (Mean Opinion Score) of 3.65, which is comparable with the 3.78 MOS of our baseline FastSpeech2.

D. Ablation Studies

TABLE V
ABLATION STUDIES ON THE AUDIOCAPS DATASET.

Setting	FAD↓	KL↓	CLAP↑
AudioLDM 2	1.67	**1.01**	0.249
a. *w/o Joint finetuning*	2.24	1.07	0.234
b. *w/o CLAP embedding (GPT)*	2.48	1.07	0.245
c. *w/o FLAN-T5 embedding (GPT)*	2.73	1.05	**0.250**
d. *w/o FLAN-T5 crossattn (T-UNet)*	**1.38**	1.30	0.211
e. *w/o CLAP and FLAN-T5 (GPT)*	2.11	1.06	0.213

In order to validate our design choices of *AudioLDM 2*, we conducted a series of ablation studies on the text-to-audio generation task on the AudioCaps dataset. The results are shown in Table V. When the joint finetuning process between the GPT-2 model and the latent diffusion model was disabled (*a*), thereby only optimizing them separately, all three evaluation metrics exhibited a marked deterioration, suggesting joint finetuning is helpful for the GPT-2 model to better cooperate with the LDM model. The GPT-2 model accepts inputs from both the CLAP and FLAN-T5 modules for text-to-audio generation. The removal of either module resulted in a degradation of the evaluation metrics (*b-c*). However, the CLAP score was improved when only the CLAP module was used as an input (*c*).

This improvement is likely due to the conditioning directly matching the evaluation metric. The removal of the cross-attention mechanism in the T-UNet model (*d*), which accepts the FLAN-T5 embeddings, led to a significant degradation in both the KL divergence and CLAP scores. However, it improved the FAD score, from 1.67 to 1.38. These results indicate that while AudioMAE conditioning alone can achieve better FAD, the use of FLAN-T5 conditioning provides additional language semantic information that assists the learning of the audio and text relationships. Besides, we study the effect of removing both CLAP and FLAN-T5 representations and directly use text as input to the GPT-2 model to predict LOA (*e*). The experimental result shows that our model in this setting maintains competitive performance with an FAD of 2.11 and KL of 1.06. However, the CLAP score exhibits a noticeable degradation, which indicates that the CLAP and FLAN-T5 representations can potentially improve the relationship between the text and the generated audio.

VI. Conclusion and Future Works

In this paper, we have presented *AudioLDM 2* for audio generation, achieving state-of-the-art or comparative performance on text-to-audio, text-to-music, and text-to-speech generation tasks. As a universal audio representation, the language of audio (LOA) enables self-supervised pre-training of the latent diffusion model, providing a robust foundation for the audio generation task. We further demonstrate the versatility of our proposed method by performing audio in-context learning. *AudioLDM 2* opens doors for future works on audio generation from a unified perspective. Future work includes enabling the multi-task learning of the GPT-2 model to generate audio, music, and speech simultaneously with a single model. Additionally, we plan to investigate more effective representations for the "language of audio" by exploring the integration of other audio self-supervised models, such as HuBERT [61] and wav2vec [56], into our system.

Acknowledgments

References

[1] Y. Cao, S. Li, Y. Liu, Z. Yan, Y. Dai, P. S. Yu, and L. Sun, "A comprehensive survey of AI-generated content: A history of generative AI from GAN to ChatGPT," *arXiv preprint:2303.04226*, 2023.

[2] 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:2205.04421*, 2022.

[3] F. Kreuk, G. Synnaeve, A. Polyak, U. Singer, A. Défossez, J. Copet, D. Parikh, Y. Taigman, and Y. Adi, "AudioGen: Textually guided audio generation," *International Conference on Learning Representations*, 2022.

[4] 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," *International Conference on Machine Learning*, 2023.

[5] D. Zhang, S. Li, X. Zhang, J. Zhan, P. Wang, Y. Zhou, and X. Qiu, "SpeechGPT: Empowering large language models with intrinsic cross-modal conversational abilities," *arXiv preprint:2305.11000*, 2023.

[6] S. Forsgren and H. Martiros, "Riffusion: Stable diffusion for real-time music generation, 2022," *URL https://riffusion.com/about*, vol. 6, 2022.

[7] X. Liu, Z. Zhu, H. Liu, Y. Yuan, M. Cui, Q. Huang, J. Liang, Y. Cao, Q. Kong, M. D. Plumbley *et al.*, "WavJourney: Compositional audio creation with large language models," *arXiv preprint:2307.14335*, 2023.

[8] A. Agostinelli, T. I. Denk, Z. Borsos, J. Engel, M. Verzetti, A. Caillon, Q. Huang, A. Jansen, A. Roberts, M. Tagliasacchi *et al.*, "MusicLM: Generating music from text," *arXiv preprint:2301.11325*, 2023.

[9] R. Bresin, A. de Witt, S. Papetti, M. Civolani, and F. Fontana, "Expressive sonification of footstep sounds," *Proceedings of Interactive Sonification Workshop*, 2010.

[10] J. Engel, L. Hantrakul, C. Gu, and A. Roberts, "DDSP: Differentiable digital signal processing," *International Conference on Learning Representations*, 2020.

[11] Y. Ren, C. Hu, X. Tan, T. Qin, S. Zhao, Z. Zhao, and T. Liu, "Fastspeech 2: Fast and high-quality end-to-end text to speech," in *International Conference on Learning Representations*, 2021.

[12] D. Herremans and E. Chew, "MorpheuS: Automatic music generation with recurrent pattern constraints and tension profiles," in *Proceedings of IEEE TENCON*, 2016, pp. 282–285.

[13] M. W. Lam, Q. Tian, T. Li, Z. Yin, S. Feng, M. Tu, Y. Ji, R. Xia, M. Ma, X. Song *et al.*, "Efficient neural music generation," *arXiv preprint:2305.15719*, 2023.

[14] 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 Transactions on Audio, Speech, and Language Processing*, vol. 31, pp. 1720–1733, 2023.

[15] 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," *International Conference on Machine Learning*, 2023.

[16] H. Liu, Q. Kong, Q. Tian, Y. Zhao, D. Wang, C. Huang, and Y. Wang, "VoiceFixer: Toward general speech restoration with neural vocoder," *arXiv preprint:2109.13731*, 2021.

[17] A. Baevski, W.-N. Hsu, Q. Xu, A. Babu, J. Gu, and M. Auli, "Data2Vec: A general framework for self-supervised learning in speech, vision and language," in *International Conference on Machine Learning*, 2022, pp. 1298–1312.

[18] R. Girdhar, A. El-Nouby, Z. Liu, M. Singh, K. V. Alwala, A. Joulin, and I. Misra, "ImageBind: One embedding space to bind them all," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 15 180–15 190.

[19] Q. Kong, K. Chen, H. Liu, X. Du, T. Berg-Kirkpatrick, S. Dubnov, and M. D. Plumbley, "Universal source separation with weakly labelled data," *arXiv preprint:2305.07447*, 2023.

[20] Y. Okamoto, K. Imoto, S. Takamichi, R. Yamanishi, T. Fukumori, Y. Yamashita *et al.*, "Onoma-to-wave: Environmental sound synthesis from onomatopoeic words," *Transactions on Signal and Information Processing*, vol. 11, no. 1, 2022.

[21] H. Xu, J. Li, A. Baevski, M. Auli, W. Galuba, F. Metze, C. Feichtenhofer *et al.*, "Masked autoencoders that listen," *Advances in Neural Information Processing Systems*, 2022.

[22] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," 2019.

[23] R. Rombach, A. Blattmann, D. Lorenz, P. Esser, and B. Ommer, "High-resolution image synthesis with latent diffusion models," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 10 684–10 695.

[24] W. X. Zhao, K. Zhou, J. Li, T. Tang, X. Wang, Y. Hou, Y. Min, B. Zhang, J. Zhang, Z. Dong *et al.*, "A survey of large language models," *arXiv preprint:2303.18223*, 2023.

[25] N. Zeghidour, A. Luebs, A. Omran, J. Skoglund, and M. Tagliasacchi, "SoundStream: An end-to-end neural audio codec," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 30, pp. 495–507, 2021.

[26] Z. Borsos, R. Marinier, D. Vincent, E. Kharitonov, O. Pietquin, M. Sharifi, D. Roblek, O. Teboul, D. Grangier, M. Tagliasacchi *et al.*, "AudioLM: A language modeling approach to audio generation," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 42, pp. 2523–2544, 2023.

[27] C. D. Kim, B. Kim, H. Lee, and G. Kim, "AudioCaps: Generating captions for audios in the wild," in *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2019, pp. 119–132.

[28] K. Ito and L. Johnson, "The LJSpeech dataset," https://keithito.com/LJ-Speech-Dataset/, 2017.

[29] R. Sheffer and Y. Adi, "I hear your true colors: Image guided audio generation," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2023.

[30] V. Iashin and E. Rahtu, "Taming visually guided sound generation," in *British Machine Vision Conference*, 2021.

[31] V. Popov, I. Vovk, V. Gogoryan, T. Sadekova, and M. Kudinov, "Grad-TTS: A diffusion probabilistic model for text-to-speech," in *International Conference on Machine Learning*, 2021, pp. 8599–8608.

[32] J. Kim, S. Kim, J. Kong, and S. Yoon, "Glow-TTS: A generative flow for text-to-speech via monotonic alignment search," *Advances in Neural Information Processing Systems*, pp. 8067–8077, 2020.

[33] Q. Huang, D. S. Park, T. Wang, T. I. Denk, A. Ly, N. Chen, Z. Zhang, Z. Zhang, J. Yu, C. Frank *et al.*, "Noise2Music: Text-conditioned music generation with diffusion models," *arXiv preprint:2302.03917*, 2023.

[34] J. Copet, F. Kreuk, I. Gat, T. Remez, D. Kant, G. Synnaeve, Y. Adi, and A. Défossez, "Simple and controllable music generation," *arXiv preprint:2306.05284*, 2023.

[35] Y.-A. Chung, Y. Zhang, W. Han, C.-C. Chiu, J. Qin, R. Pang, and Y. Wu, "W2V-Bert: Combining contrastive learning and masked language modeling for self-supervised speech pre-training," in *IEEE Automatic Speech Recognition and Understanding Workshop*. IEEE, 2021, pp. 244–250.

[36] J. Ho, A. Jain, and P. Abbeel, "Denoising diffusion probabilistic models," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 6840–6851.

[37] Y. Song, J. Sohl-Dickstein, D. Kingma, A. Kumar, S. Ermon, and B. Poole, "Score-based generative modeling through stochastic differential equations," in *International Conference on Learning Representations*, 2021.

[38] P. Dhariwal and A. Nichol, "Diffusion models beat GANs on image synthesis," in *Advances in Neural Information Processing Systems*, 2021.

[39] A. Ramesh, P. Dhariwal, A. Nichol, C. Chu, and M. Chen, "Hierarchical text-conditional image generation with CLIP latents," *arXiv preprint:2204.06125*, 2022.

[40] C. Saharia, W. Chan, S. Saxena, L. Li, J. Whang, E. Denton, S. K. S. Ghasemipour, B. K. Ayan, S. S. Mahdavi, R. G. Lopes, T. Salimans, J. Ho, D. J. Fleet, and M. Norouzi, "Photorealistic text-to-image diffusion models with deep language understanding," *arXiv preprint:2205.11487*, 2022.

[41] C. Saharia, J. Ho, W. Chan, T. Salimans, D. J. Fleet, and M. Norouzi, "Image super-resolution via iterative refinement," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 45, no. 4, pp. 4713–4726, 2022.

[42] N. Chen, Y. Zhang, H. Zen, R. Weiss, M. Norouzi, and W. Chan, "WaveGrad: Estimating gradients for waveform generation," in *International Conference on Learning Representations*, 2021.

[43] Z. Kong, W. Ping, J. Huang, K. Zhao, and B. Catanzaro, "DiffWave: A versatile diffusion model for audio synthesis," in *International Conference on Learning Representations*, 2021.

[44] Y. Leng, Z. Chen, J. Guo, H. Liu, J. Chen, X. Tan, D. Mandic, L. He, X.-Y. Li, T. Qin *et al.*, "BinauralGrad: A two-stage conditional diffusion probabilistic model for binaural audio synthesis," *Advances in Neural Information Processing Systems*, 2022.

[45] U. Singer, A. Polyak, T. Hayes, X. Yin, J. An, S. Zhang, Q. Hu, H. Yang, O. Ashual, O. Gafni *et al.*, "Make-a-video: Text-to-video generation without text-video data," in *International Conference on Learning Representations*, 2022.

[46] J. Ho, W. Chan, C. Saharia, J. Whang, R. Gao, A. Gritsenko, D. P. Kingma, B. Poole, M. Norouzi, D. J. Fleet, and T. Salimans, "Imagen video: High definition video generation with diffusion models," *arXiv preprint:2210.02303*, 2022.

[47] Z. Chen, Y. Wu, Y. Leng, J. Chen, H. Liu, X. Tan, Y. Cui, K. Wang, L. He, S. Zhao, J. Bian, and D. Mandic, "ResGrad: Residual denoising diffusion probabilistic models for text to speech," *arXiv preprint:2212.14518*, 2022.

[48] M. Lam, J. Wang, R. Huang, D. Su, and D. Yu, "Bilateral denoising diffusion models," in *International Conference on Learning Representations*, 2022.

[49] S. Lee, H. Kim, C. Shin, X. Tan, C. Liu, Q. Meng, T. Qin, W. Chen, S. Yoon, and T. Liu, "Priorgrad: Improving conditional denoising diffusion models with data-driven adaptive prior," in *International Conference on Learning Representations*, 2022.

[50] Z. Chen, X. Tan, K. Wang, S. Pan, D. Mandic, L. He, and S. Zhao, "Infergrad: Improving diffusion models for vocoder by considering inference in training," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2022.

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

[52] X. Liu, H. Liu, Q. Kong, X. Mei, M. D. Plumbley, and W. Wang, "Simple pooling front-ends for efficient audio classification," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2023.

[53] H. Liu, X. Liu, Q. Kong, W. Wang, and M. D. Plumbley, "Learning the spectrogram temporal resolution for audio classification," *arXiv preprint:2210.01719*, 2022.

[54] A. van den Oord, S. Dieleman, H. Zen, K. Simonyan, O. Vinyals, A. Graves, N. Kalchbrenner, A. Senior, and K. Kavukcuoglu, "WaveNet: A generative model for raw audio," in *ISCA Speech Synthesis Workshop*, 2016, pp. 125–125.

[55] X. Tan, T. Qin, J. Bian, T.-Y. Liu, and Y. Bengio, "Regeneration learning: A learning paradigm for data generation," *arXiv preprint:2301.08846*, 2023.

[56] 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.

[57] D. P. Kingma and M. Welling, "Auto-encoding variational bayes," *International Conference on Learning Representations*, 2014.

[58] 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 *International Conference on Learning Representations*, 2020.

[59] J. F. Gemmeke, D. P. 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.

[60] Y. Li, R. Yuan, G. Zhang, Y. Ma, X. Chen, H. Yin, C. Lin, A. Ragni, E. Benetos, N. Gyenge *et al.*, "MERT: Acoustic music understanding model with large-scale self-supervised training," *arXiv preprint:2306.00107*, 2023.

[61] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "HuBERT: Self-supervised speech representation learning by masked prediction of hidden units," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, pp. 3451–3460, 2021.

[62] S. Schneider, A. Baevski, R. Collobert, and M. Auli, "Wav2Vec: Unsupervised pre-training for speech recognition," *INTERSPEECH*, pp. 3465–3469, 2019.

[63] D. Niizumi, D. Takeuchi, Y. Ohishi, N. Harada, and K. Kashino, "Byol for audio: Self-supervised learning for general-purpose audio representation," in *International Joint Conference on Neural Networks*, 2021.

[64] J. Kong, J. Kim, and J. Bae, "HiFi-GAN: Generative adversarial networks for efficient and high fidelity speech synthesis," *Advances in Neural Information Processing Systems*, vol. 33, pp. 17 022–17 033, 2020.

[65] K. J. Piczak, "ESC: Dataset for environmental sound classification," in *Proceedings of the ACM International Conference on Multimedia*, 2015, pp. 1015–1018.

[66] L. Van der Maaten and G. Hinton, "Visualizing data using t-SNE." *Journal of Machine Learning Research*, vol. 9, 2008.

[67] Y. Qu, P. Liu, W. Song, L. Liu, and M. Cheng, "A text generation and prediction system: Pre-training on new corpora using BERT and GPT-2," in *IEEE International Conference on Electronics Information and Emergency Communication*, 2020, pp. 323–326.

[68] T. Klein and M. Nabi, "Learning to answer by learning to ask: Getting the best of GPT-2 and BERT worlds," *arXiv preprint:1911.02365*, 2019.

[69] A. M. Lamb, A. G. ALIAS PARTH GOYAL, Y. Zhang, S. Zhang, A. C. Courville, and Y. Bengio, "Professor forcing: A new algorithm for training recurrent networks," *Advances in Neural Information Processing Systems*, vol. 29, 2016.

[70] S. Masoudnia and R. Ebrahimpour, "Mixture of experts: A literature survey," *Artificial Intelligence Review*, vol. 42, pp. 275–293, 2014.

[71] 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 *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2023.

[72] H.-H. Wu, O. Nieto, J. P. Bello, and J. Salomon, "Audio-text models do not yet leverage natural language," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2023.

[73] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, E. Li, X. Wang, M. Dehghani, S. Brahma *et al.*, "Scaling instruction-finetuned language models," *arXiv preprint:2210.11416*, 2022.

[74] 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," *The Journal of Machine Learning Research*, vol. 21, no. 1, pp. 5485–5551, 2020.

[75] X. Tan, *Neural Text-to-Speech Synthesis*, ser. Artificial Intelligence: Foundations, Theory, and Algorithms. Springer Singapore, 2023.

[76] D. P. Kingma and M. Welling, "Auto-encoding variational Bayes," *arXiv preprint:1312.6114*, 2013.

[77] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in Neural Information Processing Systems*, vol. 30, 2017.

[78] J. Ho and T. Salimans, "Classifier-free diffusion guidance," in *NeurIPS Workshop on Deep Generative Models and Downstream Applications*, 2021.

[79] A. Q. Nichol, P. Dhariwal, A. Ramesh, P. Shyam, P. Mishkin, B. Mcgrew, I. Sutskever, and M. Chen, "GLIDE: Towards photorealistic image generation and editing with text-guided diffusion models," in *International Conference on Machine Learning*, 2022, pp. 16 784–16 804.

[80] 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:2303.17395*, 2023.

[81] H. Chen, W. Xie, A. Vedaldi, and A. Zisserman, "VGGSound: A large-scale audio-visual dataset," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2020, pp. 721–725.

[82] M. Defferrard, K. Benzi, P. Vandergheynst, and X. Bresson, "FMA: A dataset for music analysis," in *International Society for Music Information Retrieval Conference*, 2017.

[83] T. Bertin-Mahieux, D. P. Ellis, B. Whitman, and P. Lamere, "The million song dataset," *International Society for Music Information Retrieval Conference*, pp. 591–596, 2011.

[84] G. Chen, S. Chai, G. Wang, J. Du, W. Q. Zhang, C. Weng, D. Su, D. Povey, J. Trmal, J. Zhang *et al.*, "GigaSpeech: An evolving, multi-domain asr corpus with 10,000 hours of transcribed audio," in *INTERSPEECH*, 2021, pp. 4376–4380.

[85] S. Doh, M. Won, K. Choi, and J. Nam, "Toward universal text-to-music retrieval," *arXiv preprint:2211.14558*, 2022.

[86] S. Hershey, S. Chaudhuri, D. P. Ellis, J. F. Gemmeke, A. Jansen, R. C. Moore, M. Plakal, D. Platt, R. A. Saurous, B. Seybold *et al.*, "CNN architectures for large-scale audio classification," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2017, pp. 131–135.

[87] K. Koutini, J. Schlüter, H. Eghbal-Zadeh, and G. Widmer, "Efficient training of audio transformers with patchout," *INTERSPEECH*, pp. 2753–2757, 2021.

[88] R. Likert, "A technique for the measurement of attitudes." *Archives of Psychology*, 1932.

[89] Z. Chen, N. Kanda, J. Wu, Y. Wu, X. Wang, T. Yoshioka, J. Li, S. Sivasankaran, and S. E. Eskimez, "Speech separation with large-scale self-supervised learning," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, 2023.

[90] J. Song, C. Meng, and S. Ermon, "Denoising diffusion implicit models," in *International Conference on Learning Representations*, 2020.

[91] I. Loshchilov and F. Hutter, "Decoupled weight decay regularization," in *International Conference on Learning Representations*, 2019.

[92] J. Huang, Y. Ren, R. Huang, D. Yang, Z. Ye, C. Zhang, J. Liu, X. Yin, Z. Ma, and Z. Zhao, "Make-An-Audio 2: Temporal-enhanced text-to-audio generation," *arXiv preprint:2305.18474*, 2023.

[93] F. Schneider, Z. Jin, and B. Schölkopf, "Mousai: Text-to-music generation with long-context latent diffusion," *arXiv preprint:2301.11757*, 2023.

Haohe Liu is a PhD student at the Centre for Vision Speech and Signal Processing (CVSSP), University of Surrey, UK, and previously obtained his B.Eng degree from Northwestern Polytechnical University, Xi'an, China in 2020. His past research has contributed to the fields of audio quality enhancement, audio generation, source separation, and audio recognition. Haohe is mostly known for developing AudioLDM for text-to-audio generation, generating wide attention in the open-source community. His work as the primary author has been accepted in top journals and conferences such as TPAMI, TASLP, ICML, AAAI, ICASSP, and INTERSPEECH. Notable recent projects include AudioLDM, VoiceFixer, AudioSR, and NaturalSpeech.
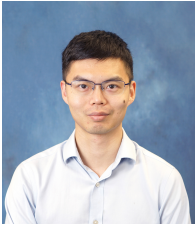
Yi Yuan received a B.Eng. degree from the University of Sydney, Sydney, Australia, and the Harbin Engineering University, Harbin, China, in 2021, the M.S. degree in Artificial Intelligence from the University of Surrey, Guildford, UK, in 2022. He is currently working toward a PhD degree in Vision, Speech and Signal Processing with the University of Surrey. His research focuses on the area of deep-learning-based audio generation. In 2023, he achieved the top-1 ranking with DCASE data challenge task 7.

Xubo Liu is a PhD student at the Centre for Vision, Speech, and Signal Processing (CVSSP) at the University of Surrey, UK. His PhD is working on multimodal learning for audio and language, focusing on the understanding, separation, and generation of audio signals in tandem with natural language. He has co-authored over 40 papers in top conferences such as CVPR, ICML, AAAI, EMNLP, ICASSP, Interspeech, etc. He has organized the "Multimodal Learning for Audio and Language" special session at EUSIPCO 2023. He is organizing the "Language-Queried Audio Source Separation" challenge on DCASE 2024. He is a frequent reviewer for TASLP, CVPR, EMNLP, ICASSP, Interspeech, and MLSP.
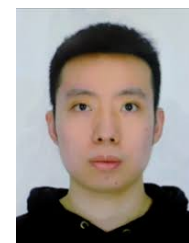
Xinhao Mei is currently pursuing a PhD degree at the Centre for Vision, Speech, and Signal Processing at the University of Surrey in Guildford, U.K. His research primarily focuses on multimodal deep learning and, more specifically, audio understanding and generation.

Qiuqiang Kong is currently an assistant professor at the EE department of the Chinese University of Hong Kong. He received his Ph.D. degree from the University of Surrey, Guildford, UK, in 2019. Following his Ph.D., he joined ByteDance as a research scientist. His research topic includes the classification, detection, separation, and generation of general sounds and music. He was the top 2% scientist in 2021 in "Updated science-wide author databases of standardized citation indicators". He was known for developing pretrained audio neural networks (PANNs) for audio tagging and was awarded the IEEE SPS Young Author Best Paper in 2023. He has co-authored over 50 papers in journals and conferences, including IEEE/ACM Transactions on Audio, Speech, and Language Processing (TASLP), etc. He assisted with organizing the LVA-ICA 2018 in Guildford, UK and the DCASE 2018 Workshop in Woking, UK.

Qiao Tian is the research scientist of the SAMI (Speech, Audio & Music Intelligence) team at the ByteDance AI Research lab. He has been focusing on developing deep learning-based technology for audio applications.

Yuping Wang graduated with a master's degree from the Department of Information and Engineering, University of Science and Technology of China. He obtained a bachelor's degree in 2003 and a master's degree in 2006. Since graduation, he has been engaged in research on audio generation-related technologies and currently works in the speech department at ByteDance.

Wenwu Wang was born in Anhui, China. He received the B.Sc., M.E., and the Ph.D. degrees from Harbin Engineering University, China, in 1997, 2000, and 2002, respectively. He then worked with King's College London, Cardiff University, Tao Group Ltd. (now Antix Labs Ltd.), and Creative Labs, before joining University of Surrey, U.K., in May 2007, where he is currently a Professor in Signal Processing and Machine Learning, University of Surrey, UK. He is also an AI Fellow at the Surrey Institute for People Centred Artificial Intelligence. His current research interests include signal processing, machine learning and perception, artificial intelligence, machine audition (listening), and statistical anomaly detection. He has (co-)authored over 300 papers in these areas. He has been recognized as a (co-)author or (co-)recipient of more than 15 awards, including the 2022 IEEE Signal Processing Society Young Author Best Paper Award, ICAUS 2021 Best Paper Award, DCASE 2020 and 2023 Judge's Award, DCASE 2019 and 2020 Reproducible System Award, and LVA/ICA 2018 Best Student Paper Award. He is an Associate Editor (2020-2025) for IEEE/ACM Transactions on Audio Speech and Language Processing and Associate Editor (2024-2026) for IEEE Transactions on Multimedia. He was a Senior Area Editor (2019-2023) and Associate Editor (2014-2018) for IEEE Transactions on Signal Processing. He is the elected Chair (2023-2024) of IEEE Signal Processing Society (SPS) Machine Learning for Signal Processing Technical Committee, a Board Member (2023-2024) of IEEE SPS Technical Directions Board, the Chair (2025-2027) and Vice Chair (2022-2024) of the EURASIP Technical Area Committee on Acoustic Speech and Music Signal Processing, an elected Member (2021-2026) of the IEEE SPS Signal Processing Theory and Methods Technical Committee. He was a Satellite Workshop Co-Chair for INTERSPEECH 2022, a Publication Co-Chair for IEEE ICASSP 2019, Local Arrangement Co-Chair of IEEE MLSP 2013, and Publicity Co-Chair of IEEE SSP 2009. He is a Satellite Workshop Co-Chair for IEEE ICASSP 2024, Special Session Co-Chair of IEEE MLSP 2024, and Technical Program Co-Chair of IEEE MLSP 2025.
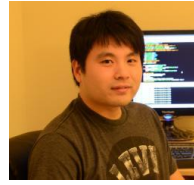
Yuxuan Wang obtained his PhD degree from the Ohio State University in 2015 and is currently the head of SAMI (the Speech, Audio and Music Intelligence team) at ByteDance, where he leads the team to conduct cutting-edge research and development to serve content understanding and content creation. Yuxuan is known for co-founding the Tacotron project, leading a paradigm shift in the text-to-speech field. Yuxuan has also made significant contributions to the fields of robust speech frontend and multi-microphone algorithms.

Mark D. Plumbley (Fellow, IEEE) received the B.A. (Hons.) degree in electrical sciences and the Ph.D. degree in neural networks from University of Cambridge, Cambridge, U.K., in 1984 and 1991, respectively. After his PhD degree, he became a Lecturer at King's College London, London, before moving to Queen Mary University of London, London, in 2002. He subsequently became a Professor and a Director of the Centre for Digital Music, before joining the University of Surrey in 2015 as a Professor of signal processing. He is known for his work on the analysis and processing of audio and music, using a wide range of signal-processing techniques, including matrix factorization, sparse representations, and deep learning. He is the co-editor of the recent book on Computational Analysis of Sound Scenes and Events, and Co-Chair of the recent DCASE 2018 Workshop on Detection and Classifications of Acoustic Scenes and Events. He is a Member of the IEEE Signal Processing Society Technical Committee on Signal Processing Theory and Methods and a Fellow of the IET.