TETA: Temporal-Enhanced Text-to-Audio Generation

Anonymous ACL submission

Abstract

Large diffusion models have been successful in text-to-audio (T2A) synthesis tasks, but they often suffer from common issues such as semantic misalignment and poor temporal consistency due to limited natural language understanding and data scarcity. Additionally, 2D spatial structures widely used in T2A works lead to unsatisfactory audio quality when generating variable-length audio samples since they ignore the time-frequency structure in the mel-spectrogram. To address these challenges, we propose TETA, a latent diffusion-based T2A method. Our approach includes several techniques to improve semantic alignment and temporal consistency: Firstly, we use pre-trained large language models (LLMs) to parse the text into structured <event & order> pairs for better temporal information capture. We also introduce another structured-text encoder to aid in learning semantic alignment during the diffusion denoising process. To improve the performance of variable length generation and enhance the temporal information extraction, we design a feed-forward Transformer-based diffusion denoiser. Finally, we use LLMs to augment and transform a large amount of audio-label data into audio-text datasets to alleviate the problem of scarcity of temporal data. Extensive experiments show that our method outperforms baseline models in both objective and subjective metrics, and achieves significant gains in temporal information understanding, semantic consistency, and sound quality. Our demos are available at https://teta2023.github.io.

1 Introduction

Deep generative learning models (Goodfellow et al., 2020; Kingma and Dhariwal, 2018; Ho et al., 2020) have revolutionized the creation of digital content, enabling creators with no professional training to produce high-quality images (Rombach et al., 2022; Saharia et al., 2022), vivid videos (Hong et al., 2022; Singer et al., 2022), diverse styles of voice (Huang et al., 2022), and meaningful long textual spans (Zhang et al., 2022; OpenAI, 2023). Text-to-audio synthesis (T2A) is a subcategory of generative tasks that aims to generate natural and accurate audio by taking text prompts as input. T2A can be useful in generating desired sound effects, music and speech, and can be applied to various applications like movie sound effects making, virtual reality, game development, and audio editing.

Thanks to the development of text-to-image synthesis (T2I) methods, researchers have successfully extended similar approaches to the text-to-audio synthesis domain (Huang et al., 2023; Liu et al., 2023; Yang et al., 2023; Kreuk et al., 2023). The success of these methods has opened up numerous opportunities for generating high-quality audio content from text. T2A systems typically use a text encoder to encode the audio's text input as condition embedding, then employ diffusion models (Huang et al., 2023; Liu et al., 2023; Yang et al., 2023) to synthesis mel-spectrograms, or utilize auto-regressive models (Kreuk et al., 2023) to synthesis raw waveform data based on the condition embedding. However, previous T2A methods have some common issues: 1) **Temporal disorder**: when the text input is complex, with multiple objects and temporal relationships between them, the generated audios often suffer from semantic misalignment and temporal disorder. For instance, audio captions such as "The sound of A, followed by the sound of B" may result in audios where A and B overlap throughout, or B comes before A, or even only one sound is synthesized. 2) **Poor variable-length results**: previous diffusion-based works (Huang et al., 2023; Liu et al., 2023) adopt the U-Net structure of 2D convolution and spatial transformer stacking as the backbone of diffusion denoiser, which is typically trained with fixed-length audios. Consequently, they generate suboptimal results when synthesizing audio sequences

of varying lengths compared to those of the training data. 2D spatial structures are not good at extracting temporal information since they treat the time axis and frequency axis equally, ignoring the time-frequency mel-spectrogram structure. 3) **Insufficient temporal paired data**: previous works use simple rule-based augmentation methods (Elizalde et al., 2022; Kreuk et al., 2023) to create temporally aligned text-audio paired data from audio-label datasets. However, these patterns are overly simplistic and can hinder the model's ability to generalize to real-world sentences.

In this paper, we propose a novel temporal-enhanced text-to-audio generation framework. The temporal information can be better handled by our method in the following ways: 1) To address the semantic misalignment and temporal disorder, we use a pre-trained LLM to extract the audio caption's temporal information and parse the origin caption into structured <event & order> pairs with proper prompts. To encode the structured pairs better, we introduce another structured-text encoder that takes the structured pairs as its input to aid in learning semantic alignment during the diffusion denoising process. In this way, we relieve the text encoder's burden of recognizing events with the corresponding temporal information and enable the T2A system to model the timing information of the events more effectively. 2) To improve the generation quality of variable-length audio and enhance the temporal information understanding, we replace the 2D spatial structures with temporal feed-forward Transformer (Ren et al., 2019) and 1D-convolution stacks for the diffusion denoiser and support variable-length audio input in training. 3) To address the issue of insufficient temporally aligned audio-text paired dataset, we use single-labeled audio samples and their labels to compose complex audio and structured captions. We then use LLM to augment the structured caption into natural language captions.

We conduct extensive experiments on AudioCaps and Clotho datasets, which reveals that our method surpasses baseline models in both objective and subjective metrics, and achieves significant gains in understanding temporal information, maintaining semantic consistency, and enhancing sound quality. Our ablation studies further demonstrate the effectiveness of each of our techniques.

2 Related works

2.1 Text-to-image generative models

Text-to-Image Synthesis (T2I) has garnered significant attention in recent years. One pioneering work in this realm is DALL-E (Ramesh et al., 2021), which treats T2I generation as a sequence-to-sequence translation task. DALL-E employs a pre-trained VQ-VAE (Van Den Oord et al., 2017) to encode image patches to discrete codes, which are then combined with the text codes. During inference, the model generates image codes autoregressively based on the text codes. DALLE-2 (Ramesh et al., 2022) uses the CLIP (Radford et al., 2021) text encoder and two diffusion models. The first diffusion model predicts CLIP visual features based on the CLIP text feature, while the second synthesizes the image from the predicted CLIP visual features. Another famous T2I work is Imagen (Saharia et al., 2022), which utilizes the T5 encoder (Raffel et al., 2020) to extract text features, It employs a diffusion model to synthesize a low-resolution image and then applies a cascade of diffusion models for super-resolution. Latent Diffusion (Rombach et al., 2022) enhances computational efficiency by using a continuous VAE trained with a discriminator to map images from pixel space to compressed latent space. This is followed by diffusion on the latent space, which synthesizes images' latent.

2.2 Text-to-audio synthesis

Text-to-Audio Synthesis is a rising task that has seen great advances recently. Diffsound (Yang et al., 2023) uses a pre-trained VQ-VAE (Van Den Oord et al., 2017) trained on mel-spectrograms to convert audio into discrete codes, which are then used by a diffusion model to generate the audio codes. To improve its generalization ability, the authors pre-trained Diffsound on the AudioSet dataset, which contains audio files labeled with tags. And they introduce a random input masking technique to make use of these tags. AudioGen (Kreuk et al., 2023) uses a similar VQ-VAE-based approach. It encodes raw waveform data into discrete codes and employs an autoregressive model to predict audio tokens based on text features. For data augmentation, AudioGen mixes audio files and concatenates their text captions. Make-An-Audio (Huang et al., 2023), AudioLDM (Liu et al., 2023), and TANGO (Ghosal et al., 2023) are all based on the Latent Diffusion Model (LDM).

With the assumption that CLAP can map the audio and its caption to the same latent space and approximate the text features based on the audio feature, AudioLDM uses audio features extracted by the CLAP model as the condition during training and uses text features during inference. Make-An-Audio and TANGO employ text features both in the training and inference stages. To overcome data scarcity, Make-An-Audio proposes a pseudo prompt enhancement method, using pre-trained audio captioning and audio-text retrieval models to generate natural language sentences to describe the content of audio clips and then concatenate the audio clips and the generated captions by predefined templates. Due to the limited capabilities of the audio captioning model and the manual setting of text templates, the naturalness and diversity of the resulting text descriptions are limited. TANGO introduces an audio mixture method based on human auditory perception and simply concatenates captions.

3 Method

In this section, we first provide an overview of the framework of our method. We then introduce our temporal enhancement method and dual text encoder structure, which aims to capture temporal information more effectively and improve the semantic alignment between the text and audio. We then present our LLM-based augmentation method, which further enhances the generalization ability and performance of our model in terms of generating audio with high semantic correspondence. In the end, we illustrate the structure of our diffusion denoiser, which is designed to enhance the generation of variable-length audio.

3.1 Overview

Our framework overview is shown in Figure 1. Denote an audio-text pair as (a, y) where $a \in R^{T_a}$ and T_a is the waveform length. To mitigate the complexity of modeling long continuous waveform data, we first convert a to mel-spectrogram (akin to the 1-channel 2D image) $x \in R^{C_a \times T}$, where $C_a, T \ll T_a$ denote the mel-channels and the number of frames respectively. The training process includes two stages:

Training variational autoencoder The audio encoder E takes mel-spectrogram x as input and outputs compressed latent $z = E(x)$. The audio decoder D reconstructs the mel-spectrogram signals $x' = D(z)$ from the compressed representation z. VAE solves the problem of excessive smoothing in mel-spectrogram reconstruction through adversarial training with a discriminator. The training objective is to minimize the weighted sum of reconstruction loss $\mathcal{L}_{re}$, GAN loss $\mathcal{L}_{GAN}$ and KL-penalty loss $\mathcal{L}_{KL}$.

Training latent diffusion model Diffusion models (Ho et al., 2020; Rombach et al., 2022) consists of two processes. In forward process, given the latent z encoded by the VAE, diffusion model transforms z into standard Gaussian distribution by T steps, the data distribution of z_t at step t can be formulated as:

$$q\left(z_t \mid z_{t-1}\right) = \sqrt{1-\beta_t} z_{t-1} + \sqrt{\beta_t}\epsilon_t \quad (1)$$

Where $\beta_t \in [0, 1]$ is a predefined noise schedule hyper-parameter, $\epsilon_t \sim N(0, I)$ denotes the injected noise.

In the backward process, the latent diffusion model learns to reconstruct the data distribution of z with the conditional embedding $c = f_{cond}(y)$ of conditional encoder f_{cond}. The training objective of the diffusion module is to minimize the mean squared error in the noise space:

$$\mathcal{L}_\theta = \|\epsilon_\theta(z_t, t, c) - \epsilon\|_2^2, \quad (2)$$

$\epsilon \sim \mathcal{N}(0, I)$ denotes the noise, ϵ_θ is the denoising network, t is random time step. The diffusion model can be trained by optimizing ELBO (Ho et al., 2020), ensuring faithful reconstructions that match the ground-truth distribution.

To further improve the conditional generation performance, we adopt the classifier-free guidance (Ho and Salimans, 2021) technique. By jointly training a conditional and an unconditional diffusion model it can control the extent to which the condition information affects the generation at each sampling step and attain a trade-off between sample quality and diversity. At the training step, we randomly replace the audio caption with an empty string to get the empty string conditional embedding $c_\emptyset$ to train the unconditional model. During sampling, the output of the model is extrapolated further in the direction of $\epsilon_\theta(\mathbf{z}_t, t, c)$ and away from $\epsilon_\theta(\mathbf{z}_t, t, c_\emptyset)$ with the guidance scale $s \geq 1$:

$$\tilde{\epsilon}_\theta(z_t, t, c) = \epsilon_\theta(\mathbf{z}_t, t, c_\emptyset) + s \cdot (\epsilon_\theta(\mathbf{z}_t, t, c) - \epsilon_\theta(\mathbf{z}_t, t, c_\emptyset)) \quad (3)$$

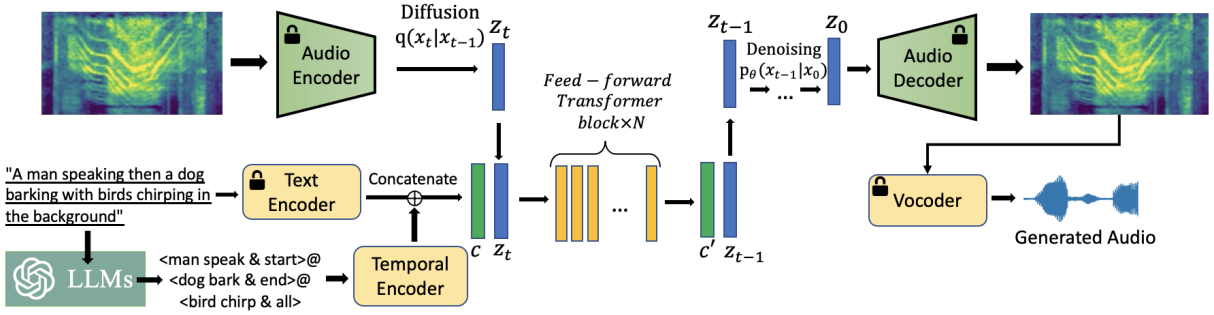

Figure 1: A high-level overview of our method. Note that modules printed with a *lock* are frozen when training the T2A model.

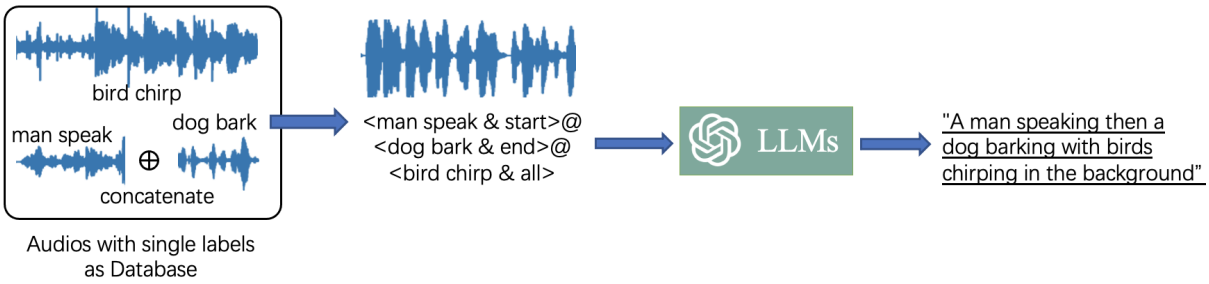

Figure 2: Overview of LLM-based data augmentation. We use single-labeled audios and their labels as a database. Composing complex audios and structured captions with these data. We then use LLM to generate diverse natural language captions by the constructed captions and appropriate prompt.

3.2 Temporal enhancement

In comparison to image data, audio data includes temporal information. A sound event can occur at any time within the audio, making audio synthesis a challenge when attempting to maintain temporal consistency. Previous approaches have encountered difficulties in dealing with captions that contain multiple sounds and complex temporal information, leading to semantic misalignment and poor temporal consistency. This can cause the generated audio to omit some sounds and produce an inaccurate temporal sequence. To address these issues, we propose the **temporal enhancement** method by parsing the original caption into structured pairs of <event & order>.

Specifically, we utilize the robust language understanding capabilities of LLMs to provide temporal knowledge. LLMs are utilized to parse the input text (the natural language audio caption) and extract structured <event & order> pairs. As illustrated in Figure 1, we use LLMs to simplify the original natural language caption and link each sound event to its corresponding order. Benefiting from enhanced temporal knowledge, the T2A model is empowered to identify sound events and corresponding temporal order. Appendix E contains further details on prompt design and additional examples of temporal enhancement.

3.3 Dual text encoders

To enhance the utilization of caption information, we propose a dual text encoder architecture consisting of a main text encoder CLAP (Elizalde et al., 2022) that takes the original natural language caption y as input, and a temporal encoder FLAN-T5 (Chung et al., 2022) which takes the structured caption y_s passed by LLM as input. The final conditional representation is expressed as:

$$c = Linear(Concat(f_{text}(y), f_{temp}(y_s))), \quad (4)$$

Where f_{text} is the main text encoder and f_{temp} is the temporal encoder. With contrastive multimodal pre-training, the CLAP has achieved excellent zero-shot performance in several downstream tasks. We freeze the weights of the main text encoder and fine-tune the temporal encoder to capture information about the temporal order of various events. As we use LLM to parse the original natural language input, some adjectives or quantifiers may be lost in this procedure, and sometimes the structured inputs' format is incorrect. Dual text encoders can avoid information loss and are more

robust in these situations. Additionally, with the frozen main text encoder, the model can maintain its generalization ability.

3.4 LLM-based data augmentation

A major challenge faced by the current T2A mission is the scarcity of data. While the T2I task benefits from billions of text-image pairs (Schuhmann et al., 2022), there are currently only around one million open-source text-audio pairs available (Huang et al., 2023). Additionally, there is a lack of data with detailed temporal annotation; many of these audios are only loosely labeled with tags instead of natural language captions. Inspired by the success of AugGPT (Dai et al., 2023) and Wavcaps (Mei et al., 2023) using LLM model to augment data, to make the most effective use of the available data, we propose an LLM-based data augmentation technique. As depicted in Figure 2, we augment audio data and its corresponding text caption as follows:

- We begin by collecting data labeled with single tags to create our event database $\mathcal{D}$. This type of data is typically cleaner and less likely to contain unwanted noise or other sounds. We can then use this data to construct more complex data based on their durations.

- Then we randomly select $N \in \{2, 3\}$ samples from $\mathcal{D}$, mix and concatenate them at random. Concatenating at random intervals or overlaps ensures that the resulting audio contains temporal information. Mixing improves the models' ability to recognize and separate different sorts of audio for creating complex compositions.

- As the resulting audio is created, we synthesize structured captions based on the occurrence time and duration of each sound event by rules. For those events that appear almost throughout the audio, we bind them with "all". While for events that only partly occur in the audio, we bind them with "start", "mid" or "end" depending on the proportion of their occurrence time points.

- Finally, we feed the structured captions into LLM with prompts to generate diverse natural language captions. The prompt to transform structured captions to natural language captions and some examples are displayed in Appendix E.

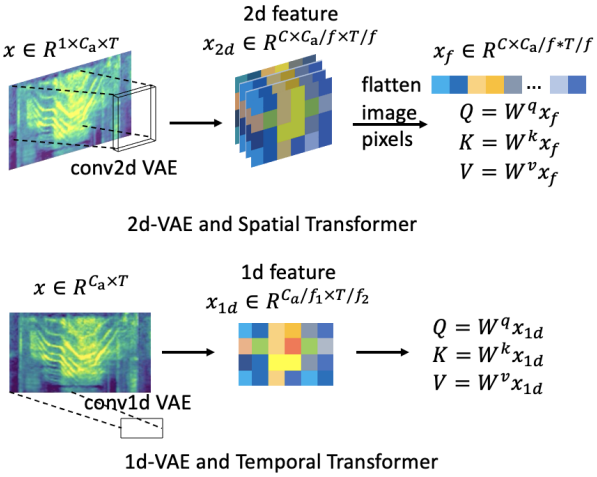

Figure 3: The illustration of differences between the 2d-VAE+spatial transformer's self-attention and 1d-VAE+temporal transformer's self-attention step in processing mel-spectrograms. We ignore the condition embedding here for simplicity. W^q, W^k, W^v are learnable projection matrices. For 2d-VAE, $W^q, W^k, W^v \in R^{D\times C}$. For 1d-VAE, $W^q, W^k, W^v \in R^{D\times C_a/f}$, D is the embedding dimension of the transformer layer. Q, K, V are used to calculate attention $Attention(Q, K, V) = softmax(QK^T/\sqrt{D})V$.

3.5 Transformer-based diffusion denoiser backbone

Previous diffusion-based work on T2A synthesis (Huang et al., 2023; Liu et al., 2023; Ghosal et al., 2023) treated the mel-spectrogram as a one-channel image similar to T2I synthesis. However, unlike images, the mel-spectrogram is not spatially translation invariant. The height of the mel-spectrogram represents the frequency domain, meaning that mel-spectrogram patches at different heights carry different meanings and should not be treated equally. The temporal translation invariant priori of 1D-convolution is more suitable as the width of mel-spectrogram represents the time domain. Furthermore, preceding approaches employ a 2D-convolution layer and spatial transformer-stacked U-Net architecture, thereby limiting the model's ability to generate variable-length audio. Illustrated in Figure 3, the spatial transformer layer, employed after the convolution layer, flattens the pixels of the 2D feature into pixel sequences. While this technique performs well with fixed-size images, it disrupts the positional information encoded by the 2D-convolution layer when the length of mel-spectrograms changes and damages the frequency-time structure constraint. Inspired by (Peebles and Xie, 2023) and (Bao et al., 2022)

that have shown U-Net is not necessary for diffusion network (Ho et al., 2020; Rombach et al., 2022) and found transformer-based (Vaswani et al., 2017) architecture can achieve better performance, we propose a modified audio VAE that uses a 1D-convolution-based model and a feed-forward Transformer-based diffusion denoiser backbone which adopts 1D-convolution and temporal transformer to improve the model's ability to generate variable-length audio.

Regarding the computational complexity of the self-attention step, while the latent of 2D-VAE audio encoder is $z = E(x) \in R^{C \times C_a/f \times T/f}$, where C is the embedding dim of latent, f is the downsampling rate, C_a and T denote the mel-channels and the number of frames of mel-spectrogram respectively, which is processed as images' height and width, f is downsampling rate. Our 1D-convolution-based audio encoder's latent is $z = E(x) \in R^{C_a/f_1 \times T/f_2}$, where C_a is taken as channel dimension rather than height when employing 2D-convolution, f_1, f_2 are downsampling rates of mel-channels and frames, respectively. Compared to the original spatial transformer, the computation complexity of the attention step in the transformer reduces from $O((C_a/f \times T/f)^2 \times D)$ to $O((T/f_2)^2 \times D)$, where D is the embedding dimension of the transformer layer.

4 Experiments

4.1 Experimental setup

Dataset. We use a combination of several datasets to train our model, including AudioCaps training set, WavCaps, AudioSet, Adobe Audition Sound Effects, Audiostock, ESC-50, FSD50K, MACS, Epidemic Sound, UrbanSound8K, WavText5Ks, TUT acoustic scene. This results in a dataset composed of **0.92 million** audio text pairs, with a total duration of approximately **3.7K hours**. More details of data filtering and preprocessing are put in Appendix A. To evaluate the performance of our models, we use the AudioCaps test set and Clotho evaluation set which contain multiple event audio samples and detailed audio captions that contain temporal information. The latter serves as a more challenging zero-shot scenario test for us, as its train set is not included in our train data.

Evaluation methods. We evaluate our models using objective and subjective metrics to assess the audio quality and text-audio alignment faithfulness. For objective evaluation, we include Frechet distance (FD), inception score (IS), Kullback–Leibler (KL) divergence, Frechet audio distance (FAD), and CLAP score. For subjective evaluation, we conduct crowd-sourced human evaluations with MOS (mean opinion score) to assess the audio quality, text-audio alignment faithfulness, and text-audio temporal alignment, scoring MOS-Q, MOS-F, and MOS-T respectively. More information about evaluation metrics and processes can be found in Appendix C.

Baseline models. To establish a standard for comparison, our study employs four baseline models, including Make-An-Audio (Huang et al., 2023) (abbreviated as MAA in Table), AudioLDM (Liu et al., 2023), TANGO (Ghosal et al., 2023) and AudioGen (Kreuk et al., 2023). We use the released version of AudioLDM-L and TANGO in Huggingface. And the released version of Make-An-Audio and AudioGen-m in Github. More model architecture and training details of our model can be found in Appendix B.

4.2 Main results

Automatic objective evaluation. The objective evaluation comparison with baseline is presented in Table 1, and we have the following observations: 1) In terms of audio quality, our model achieves better scores in FD, IS, and FAD; 2) On text-audio similarity, our model presents the comparable CLAP score with TANGO; 3) Our model achieves the lowest KL, which means the audio generated by our model has a more similar distribution of categorical labels to the ground truth audio. 4) We further compare our models' diffusion module running speed and parameter quantity with other models. Our model uses a relatively small diffusion module. Through the design of the feed-forward-transformer stacking structure, we can achieve a noteworthy acceleration while keeping the same number of parameters with Make-An-Audio. Although our model has a relatively small and faster diffusion model, we need to call ChatGPT API to parse the captions which incurs extra overhead and also influences performance. More details about the model's parameters of each component and architecture hyperparameters are attached to Appendix B.

Subjective human evaluation. The human evaluation results show significant gains of TETA with MOS-Q of 80.1, MOS-F of 78.0, and MOS-T of

Model	FD↓	IS↑	KL↓	FAD↓	CLAP↑	MOS-Q↑	MOS-F↑	MOS-T↑	Params	Speed
GroundTruth	-	-	-	-	0.671	84.3±1.41	83.8±1.66	81.6±1.61	-	-
AudioGen-m	12.42	11.65	1.43	1.85	0.604	74.6±1.77	71.5±1.91	70.6±1.45	-	1.12
MAA	13.43	11.43	1.48	2.29	0.636	73.9±1.74	72.0±1.68	71.4±1.77	160M	1.94
AudioLDM-L	23.31	8.13	1.59	1.96	0.605	71.2±1.63	69.8±1.76	67.1±2.09	739M	0.65
TANGO	26.13	8.23	1.37	1.87	**0.650**	73.7±1.69	71.6±1.81	70.8±1.80	866M	0.34
Ours	**11.45**	**11.62**	**1.25**	**1.10**	0.641	**80.1±1.60**	**78.0±1.49**	**78.7± 1.43**	160M	**2.97**

Table 1: Performance comparison on the AudioCaps dataset. All the diffusion-based models run with 100 DDIM (Song et al., 2020) steps for a fair comparison. Our model is tested with a classifier-free guidance scale of 5. We borrowed all the results from (Liu et al., 2023; Ghosal et al., 2023) and used the model released on Huggingface to test the CLAP Score. We use the released model on GitHub of Make-An-Audio (abbreviated as MAA) and AudioGen to test all the scores. Noted that Params is the diffusion module parameters, speed marks the generation speed of 10s audios per second. The experiment is conducted on one A100 GPU and the batch size (8 for diffusion networks and 32 for AudioGen) is set to make the utilization rate of GPU reach 100% when generating 10-second audio.

78.7, outperforming the current baselines. It indicates that raters prefer our model synthesis against baselines in terms of audio naturalness, text-audio semantics and temporal faithfulness.

Zero-shot evaluation. To further investigate the generalization performance of the models, we test the performance of the models on the Clotho-evaluation dataset in the zero-shot scenario. Considering audios in the Clotho-evaluation dataset have different durations, we conduct two evaluations. One is generating fixed-length audios of 10 seconds, denoted as Clotho-eval-fix. The other is to generate audio that is the same length as each piece of audio in the dataset, denoted as Clotho-eval-variable. The audio's duration in Clotho-eval-variable varies from 15s to 30s, with an average duration of 22.4 seconds. As illustrated in Table 2, our model has significantly better results than baseline models, attributed to the scalability of data usage and variable length data training.

Variable-length generation. To investigate our models' performance on variable-length audio generation, we test to generate 5-second audios and 8-second audios on AudioCaps dataset, the results are shown in Table 3. To investigate the relationship between models' performance with the duration of generated audio, we add additional experiments in Appendix D.1. We also test generating variable-length audios on the Clotho-eval dataset, as discussed in the former paragraph. It can be seen that preceding diffusion-based works, TANGO, AudioLDM and Make-An-Audio, exhibit significant performance degradation when generating audio with different lengths than the training data, while the autoregressive model AudioGen's performance remains stable in generating variable-length audio. Preceding diffusion-based works pad or truncate all the training audio data to 10 seconds, and their models are based on 2D-convolution and spatial transformer to process mel-spectrogram as images. Our model maintains high performance even when generating variable-length audio samples since it is trained on audio samples of varying lengths and utilizes 1D-convolution and temporal transformers to emphasize temporal axis information.

4.3 Ablation study

To assess the effectiveness of various designs in TETA, we conduct ablation studies on AudioCaps-test and Clotho-evaluation set. The results are presented in Table 4. The key findings are discussed below:

1D-VAE and FFT-diffusion Although 2d-convolution VAE and Unet diffusion backbone perform well trained with fixed-length data. When we tried to train them with variable-length data. They don't converge to a good result. With 1d VAE and feed-forward-transformer diffusion backbone, our model converges well when trained with variable-length data, and exhibits significant advantages in generating variable-length data.

Temporal Enhancement The results in Table 4 highlights the effectiveness of temporal enhancement. We use LLM to extract event and temporal information and create structured input as <event & order> pairs. It reduces the difficulty for the model to extract audio events and establish timing relationships from captions, which significantly improves both objective scores and sound timing modeling.

Model	Clotho-eval-fix				Clotho-eval-variale			
	FD↓	IS↑	KL↓	FAD↓	FD↓	IS↑	KL↓	FAD↓
TANGO	32.1	6.77	2.59	3.61	36.54	6.65	2.75	5.23
AudioLDM-L	28.15	6.55	2.60	4.93	24.25	7.06	2.44	4.42
MAA	22.91	8.47	2.66	3.21	26.51	8.12	2.68	5.03
AudioGen-m	23.69	7.18	2.64	2.62	24.86	6.93	2.60	2.64
Ours	**18.43**	**8.73**	**2.49**	**1.59**	**20.77**	**8.43**	**2.55**	**2.29**

Table 2: Comparison of our model and baselines on Clotho-eval datasets.

Model	Audiocaps-5s				Audiocaps-8s			
	FD↓	IS↑	KL↓	FAD↓	FD↓	IS↑	KL↓	FAD↓
TANGO	31.76	5.50	2.04	10.53	18.32	8.39	1.50	2.04
AudioLDM-L	31.97	5.66	2.39	6.79	30.95	8.65	1.91	4.91
MAA	19.30	7.83	2.13	5.61	14.25	9.85	1.64	2.27
AudioGen-m	**11.02**	**11.43**	1.67	1.63	**12.11**	**11.65**	1.52	1.63
Ours	12.40	11.10	**1.48**	**1.28**	13.20	11.15	**1.32**	**1.04**

Table 3: Comparison of our model and baselines on Audio-caps dataset. The result of generating 5-second audios and 8-second audios are denoted as Audiocaps-5s and Audiocaps-8s respectively.

Setting	Audiocaps					Clotho-eval-fix			
	FD↓	IS↑	KL↓	FAD↓	MOS-T↑	FD↓	IS↑	KL↓	FAD↓
Ours	11.45	**11.62**	1.25	**1.10**	**78.7± 1.55**	18.43	8.73	2.49	**1.59**
w/o 1d VAE + FFT diffusion	22.69	5.93	2.17	3.82	63.5± 2.21	26.59	6.92	2.67	6.02
w/o Temporal Enhancement	12.66	10.60	1.35	1.72	73.9± 1.57	21.24	8.82	2.50	2.56
w/o LLM Data Augmentation	**10.45**	11.03	**1.22**	1.25	76.1±1.87	19.75	8.63	**2.39**	2.01
w/o CLAP TextEncoder	11.91	11.07	1.29	1.59	76.9±1.69	**18.38**	**9.56**	2.43	1.94

Table 4: The ablation study of TETA. All the models are trained on variable-length data.

LLM Data Augementation We use LLM Data Augmentation to further improve the model's generalization ability and alleviate the problem of data with temporal information scarcity. The absence of LLM data augmentation results in insignificant changes in performance on the Audiocaps dataset. This can be attributed to we assigned higher data weights to the Audiocaps dataset, causing the model to become somewhat overfitted to this specific dataset. Conversely, LLM data augmentation leads to a notable improvement in performance in Clotho dataset. The MOS-T score also improves with augmentation, as the constructed data strictly follows the temporal relationships described in the captions, which helps the model learn temporal information more effectively.

Dual Text Encoder We use the frozen CLAP encoder to extract information from the original natural language caption and trainable text encoder to extract information from the parsed input. The frozen encoder provides fault-tolerance mechanisms when there are information losses and errors in the parsed input while retaining generalization capabilities. We compare the performance of the model with and without CLAP Encoder using wrongly parsed captions on the Clotho-eval dataset, the results are attached to Appendix D.2. We also compare our results when the parsed input has errors on our demo page.

5 Conclusions

In this work, we present TETA, a temporal-enhanced T2A synthesis model. With a capable LLM to extract temporal information from the natural language caption, TETA can better understand the event order in the caption and generate semantically aligned audios. Leveraging 1D-convolutional VAE and feed-forward Transformer diffusion backbone, TETA can generate variable-length audios without performance degeneration. With complex audio reconstruction and LLM-based data augmentation, TETA is endowed with the ability to understand complex temporal relationships and combinations of multiple concepts. TETA achieves the SOTA audio generation quality in both objective and subjective metrics.

6 Limitations

TETA incorporates an additional LLM for parsing the original caption, which affects both the generation performance and running speed. In temporal enhancement, we use start, mid, end, and all, it's a rough time resolution, but it works well in our experiment as the audios in our training data are not very complicated. If supplied with more complicated audio data, using the time format of order 1,2,3... can be considered. We left it as future work. Furthermore, our model lacks the capability of generating meaningful speech, the speech generated by our model is intelligible.

In terms of model evaluation, the Audiocaps and Clotho datasets pose a challenge due to their noisy nature, rendering objective metrics inadequate for assessing the model's capacity to generate pure, high-quality sound. Additionally, the performance of CLAP is limited when dealing with complex audio involving multiple sources and temporal ordering.

7 Ethics Statement

TETA improves the quality and efficiency of the audio generation, this may lead to unintended consequences such as increased unemployment for individuals in related fields such as sound engineering and radio hosting. Furthermore, there are potential concerns regarding the ethics of non-consensual voice cloning or the creation of fake media to provide misleading information.

References

Fan Bao, Chongxuan Li, Yue Cao, and Jun Zhu. 2022. All are worth words: a vit backbone for score-based diffusion models. *arXiv preprint arXiv:2209.12152*.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, and others. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Haixing Dai, Zhengliang Liu, Wenxiong Liao, Xiaoke Huang, Yihan Cao, Zihao Wu, Lin Zhao, Shaochen Xu, Wei Liu, Ninghao Liu, Sheng Li, Dajiang Zhu, Hongmin Cai, Lichao Sun, Quanzheng Li, Dinggang Shen, Tianming Liu, and Xiang Li. 2023. Auggpt: Leveraging chatgpt for text data augmentation.

Soham Deshmukh, Benjamin Elizalde, and Huaming Wang. 2022. Audio retrieval with WavText5K and CLAP training.

Konstantinos Drossos, Samuel Lipping, and Tuomas Virtanen. 2020. Clotho: An audio captioning dataset. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 736–740.

Benjamin Elizalde, Soham Deshmukh, Mahmoud Al Ismail, and Huaming Wang. 2022. Clap: Learning audio concepts from natural language supervision.

Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter. 2017. Audio set: An ontology and human-labeled dataset for audio events. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 776–780.

Deepanway Ghosal, Navonil Majumder, Ambuj Mehrish, and Soujanya Poria. 2023. Text-to-audio generation using instruction-tuned llm and latent diffusion model.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2020. Generative adversarial networks. 63(11):139–144.

Shawn Hershey, Sourish Chaudhuri, Daniel PW Ellis, Jort F Gemmeke, Aren Jansen, R Channing Moore, Manoj Plakal, Devin Platt, Rif A Saurous, Bryan Seybold, et al. 2017. Cnn architectures for large-scale audio classification. In *2017 ieee international conference on acoustics, speech and signal processing (icassp)*, pages 131–135. IEEE.

Jack Hessel, Ari Holtzman, Maxwell Forbes, Ronan Le Bras, and Yejin Choi. 2021. Clipscore: A reference-free evaluation metric for image captioning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7514–7528.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. In *Proc. of NeurIPS*.

Jonathan Ho and Tim Salimans. 2021. Classifier-free diffusion guidance. In *NeurIPS 2021 workshop on deep generative models and downstream applications*.

Wenyi Hong, Ming Ding, Wendi Zheng, Xinghan Liu, and Jie Tang. 2022. CogVideo: Large-scale pretraining for text-to-video generation via transformers.

Rongjie Huang, Jiawei Huang, Dongchao Yang, Yi Ren, Luping Liu, Mingze Li, Zhenhui Ye, Jinglin Liu, Xiang Yin, and Zhou Zhao. 2023. Make-an-audio: Text-to-audio generation with prompt-enhanced diffusion models.

Rongjie Huang, Yi Ren, Jinglin Liu, Chenye Cui, and Zhou Zhao. 2022. Generspeech: Towards style transfer for generalizable out-of-domain text-to-speech. In *Advances in Neural Information Processing Systems*.

Kevin Kilgour, Mauricio Zuluaga, Dominik Roblek, and Matthew Sharifi. 2018. Fr\'echet audio distance: A metric for evaluating music enhancement algorithms. *arXiv preprint arXiv:1812.08466*.

Chris Dongjoo Kim, Byeongchang Kim, Hyunmin Lee, and Gunhee Kim. 2019. 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, Volume 1 (Long and Short Papers)*, pages 119–132.

Durk P Kingma and Prafulla Dhariwal. 2018. Glow: Generative flow with invertible 1x1 convolutions. 31:10215–10224.

Qiuqiang Kong, Yin Cao, Turab Iqbal, Yuxuan Wang, Wenwu Wang, and Mark D Plumbley. 2020. Panns: Large-scale pretrained audio neural networks for audio pattern recognition. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 28:2880–2894.

Felix Kreuk, Gabriel Synnaeve, Adam Polyak, Uriel Singer, Alexandre Défossez, Jade Copet, Devi Parikh, Yaniv Taigman, and Yossi Adi. 2023. AudioGen: Textually Guided Audio Generation.

Haohe Liu, Zehua Chen, Yi Yuan, Xinhao Mei, Xubo Liu, Danilo Mandic, Wenwu Wang, and Mark D. Plumbley. 2023. AudioLDM: Text-to-Audio Generation with Latent Diffusion Models.

Ilya Loshchilov and Frank Hutter. 2018. Decoupled weight decay regularization. In *International conference on learning representations*.

Irene Martín-Morató and Annamaria Mesaros. 2021. What is the ground truth? reliability of multi-annotator data for audio tagging. In *2021 29th European Signal Processing Conference (EUSIPCO)*, pages 76–80.

Xinhao Mei, Chutong Meng, Haohe Liu, Qiuqiang Kong, Tom Ko, Chengqi Zhao, Mark D. Plumbley, Yuexian Zou, and Wenwu Wang. 2023. WavCaps: A ChatGPT-Assisted weakly-labelled audio captioning dataset for audio-language multimodal research.

Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen. 2016. TUT database for acoustic scene classification and sound event detection. In *2016 24th european signal processing conference (EUSIPCO)*, pages 1128–1132.

Lars Mescheder, Andreas Geiger, and Sebastian Nowozin. 2018. Which training methods for GANs do actually converge? In *International conference on machine learning*, pages 3481–3490. PMLR.

OpenAI. 2023. Gpt-4 technical report.

William Peebles and Saining Xie. 2023. Scalable diffusion models with transformers.

Karol J Piczak. 2015. ESC: Dataset for environmental sound classification. In *Proceedings of the 23rd ACM International Conference on Multimedia*, pages 1015–1018.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. 21(140):1–67.

Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. 2022. Hierarchical text-conditional image generation with clip latents. *arXiv preprint arXiv:2204.06125*.

Aditya Ramesh, Mikhail Pavlov, Gabriel Goh, Scott Gray, Chelsea Voss, Alec Radford, Mark Chen, and Ilya Sutskever. 2021. Zero-shot text-to-image generation. In *International Conference on Machine Learning*, pages 8821–8831. PMLR.

Yi Ren, Yangjun Ruan, Xu Tan, Tao Qin, Sheng Zhao, Zhou Zhao, and Tie-Yan Liu. 2019. Fastspeech: Fast, robust and controllable text to speech. *Advances in neural information processing systems*, 32.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 10684–10695.

Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily Denton, Seyed Kamyar Seyed Ghasemipour, Burcu Karagol Ayan, S Sara Mahdavi, Rapha Gontijo Lopes, et al. 2022. Photorealistic text-to-image diffusion models with deep language understanding.

J. Salamon, C. Jacoby, and J. P. Bello. 2014. A dataset and taxonomy for urban sound research. In *22nd ACM International Conference on Multimedia (ACM-MM'14)*, pages 1041–1044.

Christoph Schuhmann, Romain Beaumont, Richard Vencu, Cade Gordon, Ross Wightman, Mehdi Cherti, Theo Coombes, Aarush Katta, Clayton Mullis, Mitchell Wortsman, Patrick Schramowski, Srivatsa Kundurthy, Katherine Crowson, Ludwig Schmidt, Robert Kaczmarczyk, and Jenia Jitsev. 2022. LAION-5B: An open large-scale dataset for training next generation image-text models. ArXiv: 2210.08402 [cs.CV].

Uriel Singer, Adam Polyak, Thomas Hayes, Xi Yin, Jie An, Songyang Zhang, Qiyuan Hu, Harry Yang, Oron Ashual, Oran Gafni, et al. 2022. Make-a-video: Text-to-video generation without text-video data.

Jiaming Song, Chenlin Meng, and Stefano Ermon. 2020. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*.

Aaron Van Den Oord, Oriol Vinyals, et al. 2017. Neural discrete representation learning. *Advances in neural information processing systems*, 30.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, \Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Dongchao Yang, Jianwei Yu, Helin Wang, Wen Wang, Chao Weng, Yuexian Zou, and Dong Yu. 2023. Diffsound: Discrete Diffusion Model for Text-to-sound Generation.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

A Data details

As shown in Table 5, we collect a large-scale audio-text dataset consisting of 0.92 million audio samples with a total duration of approximately 3.7k hours. The dataset has a wide variety of sounds including music and musical instruments, sound effects, human voices, nature and living sounds, etc. For Clotho dataset, we only use its evaluation set for zero-shot testing and do not use it for training. As speech and music are the dominant classes in AudioSet, we randomly filter 95% of the samples that contain speech and music to build a more balanced dataset.

We conduct preprocessing on both text and audio data as follows:

1) We convert the sampling rate of audio to 16kHz. Prior works (Yang et al., 2023; Huang et al., 2023; Liu et al., 2023) pad or truncate the audio to a fixed length (10s), while we group audio files with similar durations to form batches to avoid excessive padding which could potentially impair model performance and slow down the training speed. This approach also allows for improved variable-length generation performance. We truncate any audio file that exceeds 20 seconds, to speed up the training process.

2) We adopt the LLM-based data augmentation method in section 3.4 to construct approximately 61k additional audio-text pairs as auxiliary data.

3) For audios without natural language annotation, we apply the pseudo prompt enhancement method from Make-An-Audio (Huang et al., 2023) to construct captions aligned with the audio.

4) We assign a lower weight to the data that is not annotated with temporal information but is abundant in quantity and diversity, such as AudioSet and WavCaps data. Specifically, we traverse the AudioCaps training set and the LLM augmented data with a probability of 50%, while randomly selecting data from all other sources with a probability of 50%. For the latter dataset, we use "<text & all>" as their structured caption.

B Experimental details

Variational autoencoder. We employed a similar VAE architecture to that of Make-An-Audio, replacing all the 2D-convolution layers with 1D-convolution layers and the spatial transformer with a temporal transformer. As detailed in Section 4.5, the output latent of VAE is $z = E(x) \in R^{C_a/f_1 \times T/f_2}$, where we choose the downsample rate of $f_1 = 4$ and $f_2 = 2$. We additionally involve R1 regularization (Mescheder et al., 2018) to better stabilize the adversarial training process. We train our VAE on 8 NVIDIA A100 GPUs with a batch size of 32 and 800k training steps on AudioSet dataset. We use the Adamw optimizer (Loshchilov and Hutter, 2018) with a learning rate of 1.44×10^{-4}. For specific differences in hyperparameters between our VAE and that of Make-An-Audio, please see Table 6.

Latent diffusion. We train our Latent Diffusion Model with on 8 NVIDIA A100 GPU with a batch size of 32 and 1.8M training steps. We use the Adam optimizer with a learning rate of 9.6×10^{-5}. For the specific hyperparameter for our latent diffusion model, please refer to Table 7.

Model parameters of each component. The params of each component in TETA are displayed in Table 8. The params comparison between our model and baselines are displayed in Table 9.

C Evaluation

C.1 subjective evaluation

To assess the generation quality, we conduct MOS (Mean Opinion Score) tests regarding audio qual-

Dataset	Hours	Type	Source
Audiocaps	109hrs	caption	(Kim et al., 2019)
WavCaps	2056hrs	caption	(Mei et al., 2023)
WavText5K	25hrs	caption	(Deshmukh et al., 2022)
MACS	48hrs	caption	(Martín-Morató and Mesaros, 2021)
Clothv2	152hrs	caption	(Drossos et al., 2020)
Audiostock	44hrs	caption	https://audiostock.net
epidemic sound	220hrs	caption	https://www.epidemicsound.com
Adobe Audition Sound Effects	26hrs	caption	https://www.adobe.com/products/audition/offers/AdobeAuditionDLCSFX.html
FSD50K	108hrs	label	https://annotator.freesound.org/fsd
ODEON_Sound_Effects	20hrs	label	https://www.paramountmotion.com/odeon-sound-effects
UrbanSound8K	9hrs	label	(Salamon et al., 2014)
ESC-50	3hrs	label	(Piczak, 2015)
filteraudioset	945hrs	multi label	(Gemmeke et al., 2017)
TUT	13hrs	label	(Mesaros et al., 2016)

Table 5: Statistics for the Datasets used in the paper.

	Make-An-Audio VAE	TETA VAE
Assume input tensor shape (for 10s audio)	(1,80,624)	(80,624)
Embed_dim	4	20
Convolution layer	Conv2D	Conv1D
Channels	128	224
Channel multiplier	1,2,2,4	1,2,4
Downsample layer position	after block 1,2	after block 1
Attention layer	spatial attention	temporal attention
Attention layer position	after block 3,4	after block 3
Output tensor shape	(4,10,78)	(20,312)

Table 6: Difference between Make-An-Audio VAE and our VAE

ity, text-audio faithfulness and text-audio temporal alignment, respectively scoring MOS-Q, MOS-F, and MOS-T.

For audio quality, the raters were explicitly instructed to "focus on examining the audio quality and naturalness." The testers were presented with audio samples and their caption and asked to rate their subjective score on a 20-100 Likert scale.

For text-audio faithfulness, human raters were shown the audio and its caption and asked to respond to the question, "Does the natural language description align with the audio faithfully?" They had to choose one of the options - "completely," "mostly," or "somewhat" on a 20-100 Likert scale.

For text-audio temporal alignment, human raters were shown the audio and its caption and asked to respond to the question, "Whether the text description contains sounds time or order information. If not then select no, if yes then score based on how the audio's sound order aligns with its caption." They had to choose one of the options - "completely," "mostly," or "somewhat" on a 20-100 Likert scale. We will filter out the audio that has been selected "no" and compute MOS-T based on the remaining audio.

Our subjective evaluation tests are crowdsourced and conducted via Amazon Mechanical Turk. These ratings are obtained independently for model samples and reference audio, and both are reported. We paid $8 to participants hourly

	TETA LDM
Input shape	(20,T)
Condition_embedding dim	1024
Feed-forward Transformer hidden_size	576
Feed-forward Transformer's Conv1d kernel size	7
Feed-forward Transformer's Conv1d padding	3
Number of Transformer heads	8
Number of Feed-forward Transformer block	8
Diffusion steps	1000

Table 7: TETA Diffusion model backbone configurations

Component	Params
VAE	213M
Diffusion Model Backbone	160M
Text Encoder	452M
Vocoder	112M
Total	937M

Table 8: The params of each component

Model	Total params	Diffusion params
AudioLDM-S	454M	185M
AudioLDM-L	1.01B	739M
Tango	1.21B	866M
Make-An-Audio	453M	160M
AudioGen-m	1.5B	-
Ours	937M	160M

Table 9: Params comparison between models.

and totally spent about $400 on participant compensation. A small subset of the generated audio samples used in the test can be found at https://teta2023.github.io/.

C.2 Objective evaluation

Fréchet Audio Distance (FAD) (Kilgour et al., 2018) is adapted from the Fréchet Inception Distance (FID) to the audio domain, it is a reference-free perceptual metric that measures the distance between the generated and ground truth audio distributions. FAD is used to evaluate the quality of generated audio. The inception Score (IS) is an effective metric that evaluates both the quality and diversity of generated audio. KL divergence is measured at a paired sample level between the generated audio and the ground truth audio, it is computed using the label distribution and is averaged as the final result. Fréchet Distance (FD) evaluates the similarity between the generated and ground truth audio distributions. FD, KL and IS are built upon an audio classifier, PANNs (Kong et al., 2020), which takes the mel-spectrogram as model input. Differently, FAD uses VGGish (Hershey et al., 2017) as an audio classifier that takes raw audio waveform as model input. CLAP score: adapted from the CLIP score (Hessel et al., 2021; Radford et al., 2021) to the audio domain and is a reference-free evaluation metric to measure audio-text alignment for this work that closely correlates with human perception.

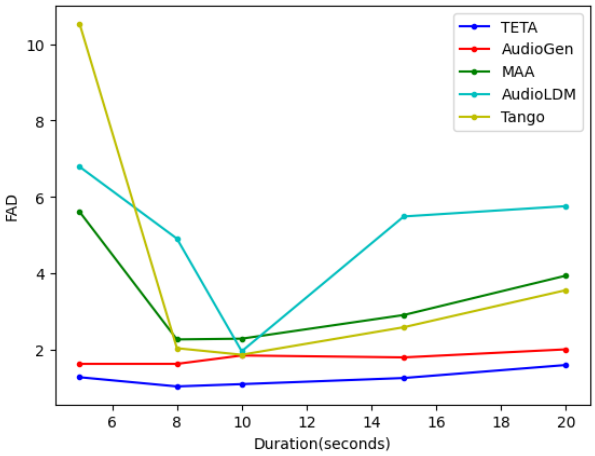

Figure 4: Fad versus duration curve of models.

D Additional Results

D.1 Variable length generation

We investigate the performance of various models in audio generation as the duration of the audio changes in Figure4. Make-An-Audio, Tango, and AudioLDM exhibit a significant variation in their generation performance across different audio durations. Specifically, as the duration of the generated

Model	FD↓	IS↑	KL↓	FAD↓	CLAP↑
TETA	72.70	6.33	2.82	6.47	0.285
w/o CLAP TextEncoder	73.64	5.02	3.01	6.54	0.189

Table 10: Performance comparison on the wrongly parsed subset.

audio deviates from the training duration of 10 seconds, the performance decreases. In contrast, TETA and AudioGen demonstrate less variation in performance across different durations, exhibiting relatively flat curves.

D.2 Performance of error parsing

We evaluate the generation performance of wrongly parsed captions (not in the format we define) on the Clotho-eval dataset. Among the 1045 captions, a subset comprising 37 structured captions was obtained after performing LLM parsing, but not correctly formatted. We generated 5 audios for each incorrectly parsed caption and tested the generation performance as shown in Table 10. For wrongly parsed captions, without CLAP text encoder leads to performance decline, particularly in clap score, which means the CLAP text encoder is helpful in information retention and fault tolerance.

E ChatGPT prompts

The prompt templates utilized for temporal enhancement to construct structure caption from the original natural language caption and for caption data augmentation are displayed in Figure 5.

Table 11 presents some instances of the original caption and ChatGPT's outcome. For text data augmentation, we construct structured caption inputs, and Table 12 exhibits examples of such inputs and ChatGPT's corresponding outputs.

F Future works

We leave the T2A system which supports speech synthesis for future work. As we have seen great potential in our LLM-based data augmentation, with elaborate prompts and merge rules, it can be used to merge speech, singing, sound events, and music to create a more universal audio scenario. Enabling the training of a model that can generate universal audios with meaningful speech and music with ideal melody. In addition, we aim to implement T2A systems that could take structured inputs as optional auxiliary inputs instead of required inputs.

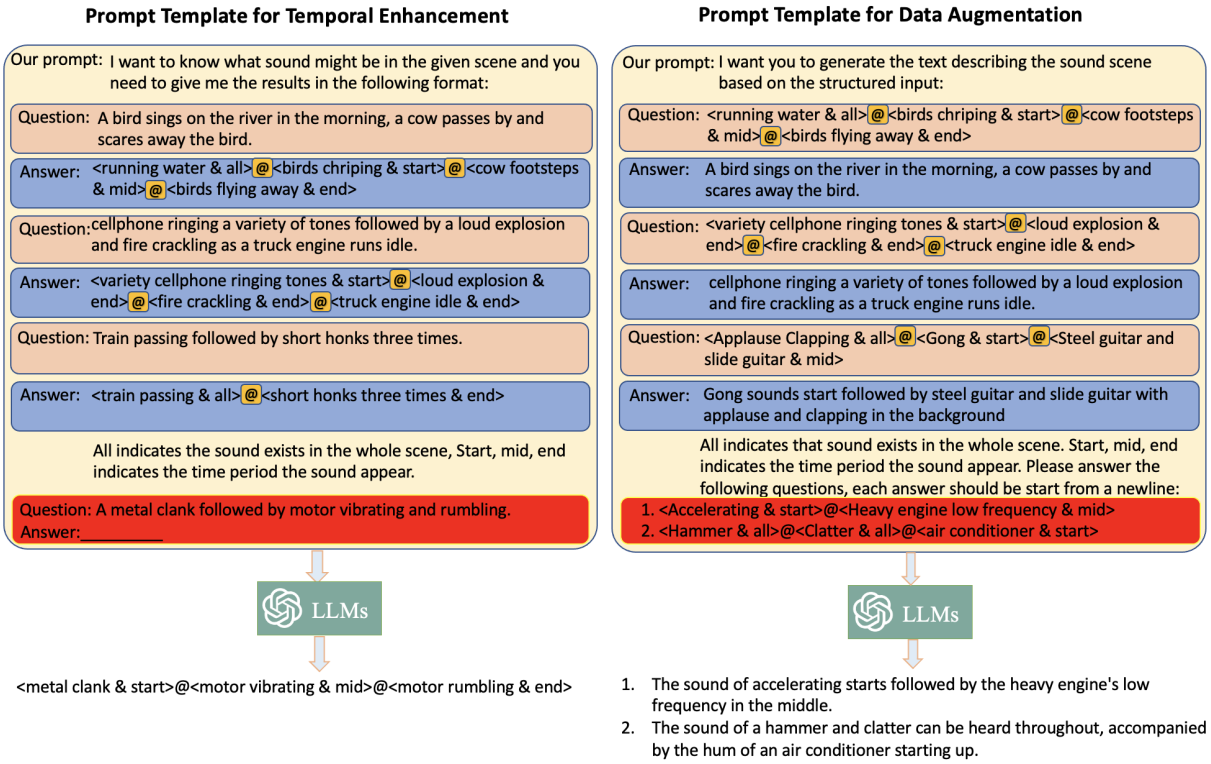

Figure 5: The prompt templates we used for temporal enhancement and data augmentation. We use the symbol '&' to split the sound event and the time order. We use the symbol '@' to split <event & order> pairs.

Natural language input	ChatGPT's output
A woman talks nearby as water pours	<woman talking& all>@<water pouring& all>
Two men exchange words, then a car engine revs followed by a siren and fade in music	<two men talking& all>@<car engine revving& start>@<siren& mid>@<music fading in& end>
A crowd is cheering and shouting, thumping occurs, an adult female speaks, and an adult male speaks	<crowd cheering and shouting& all>@<thumping& start>@<adult female speaking& mid>@<adult male speaking& end>

Table 11: Examples of using ChatGPT for temporal enhancement from AudioCaps trainset

Structured input	ChatGPT's output
<bark howl dog& start>@<typing Typewriter& mid>@<breathing& end>	A dog barks and howls while someone types on a typewriter, then the sound of breathing takes over
<crowing& all>@<car& all>@<female singing& end>	A rooster crows as a car passes by, and the scene ends with the sound of a female singing
<sneezing& all>@<bicycle bell ring& start>@<typewriter & end>	The sound of sneezing is heard throughout, with a bicycle bell ringing at the start and the sound of a typewriter at the end

Table 12: Examples of using ChatGPT for data augmentation