

ControlAudio: Tackling Text-Guided, Timing-Indicated and Intelligible Audio Generation via Progressive Diffusion Modeling

Yuxuan Jiang^{1,2,*}, Zehua Chen^{1,2,*†}, Zeqian Ju³, Yusheng Dai^{2,4}, Weibei Dou¹, Jun Zhu^{1,2,†}

¹Tsinghua University ²Shengshu AI

³University of Science and Technology of China ⁴Monash University

Abstract

Text-to-audio (TTA) generation with fine-grained control signals, *e.g.*, precise timing control or intelligible speech content, has been explored in recent works. However, constrained by data scarcity, their generation performance at scale is still compromised. In this study, we recast controllable TTA generation as a multi-task learning problem and introduce a progressive diffusion modeling approach, ControlAudio. Our method adeptly fits distributions conditioned on more fine-grained information, including text, timing, and phoneme features, through a step-by-step strategy. First, we propose a data construction method spanning both annotation and simulation, augmenting condition information in the sequence of text, timing, and phoneme. Second, at the model training stage, we pretrain a diffusion transformer (DiT) on large-scale text-audio pairs, achieving scalable TTA generation, and then incrementally integrate the timing and phoneme features with unified semantic representations, expanding controllability. Finally, at the inference stage, we propose progressively guided generation, which sequentially emphasizes more fine-grained information, aligning inherently with the coarse-to-fine sampling nature of DiT. Extensive experiments show that ControlAudio achieves state-of-the-art performance in terms of temporal accuracy and speech clarity, significantly outperforming existing methods on both objective and subjective evaluations. Demo samples are available at: <https://control-audio.github.io/Control-Audio/>.

1 Introduction

Text-to-audio (TTA) generation systems aim at synthesizing high-fidelity audio samples that are consistent with the given natural language description, *e.g.*, "A bird is chirping" (Liu et al., 2023;

Ghosal et al., 2023; Huang et al., 2023; Evans et al., 2024a). Recent efforts are exploring more fine-grained control for TTA systems, which can be categorized into two main classifications. The first group adds precise timing control, *e.g.*, "A bird is chirping, at 2-5 seconds", with innovations spanning conditioning techniques (Wang et al., 2025d; Xie et al., 2024) and training-free latent manipulation (Jiang et al., 2025). The second group works on intelligible audio generation, *e.g.*, "A bird is chirping, and a man is saying: 'it's a very sunny day'", by introducing additional modules to encode both audio and speech semantic information (Lee et al., 2024b; Jung et al., 2025). However, as expensive to collect large-scale text-audio datasets with precise timing and speech information, their controllable generation performance at scale remains limited, and none of the prior work explores *timing-controlled and intelligible TTA generation*, *e.g.*, "A bird is chirping, at 0-5 seconds, and then a man is saying: 'it's a very sunny day', at 7-10 seconds", within a unified framework.

In this work, we propose ControlAudio, a progressive diffusion modeling approach to progressively capture the target distribution conditioned on fine-grained information, (text, timing, phoneme), enabling controllable TTA generation at scale. Our designs cover data construction and representation, model training, as well as guided sampling, each of which progressively integrates more fine-grained condition information, thereby expanding controllability at scale. In data construction, we collect large-scale ⟨text, audio⟩ pairs, and then construct more expensive datasets, ⟨text, timing, audio⟩ and ⟨text, timing, phoneme, audio⟩, with both annotation and simulation methods, predefining the target distribution of each training stage. For the representation of text and timing information, we develop a structural prompt, enabling a pre-trained text encoder to precisely encode them without fine-tuning. Given the timing indication, namely the duration

*Equal Contribution.

†Corresponding Authors: Zehua Chen and Jun Zhu.

of the speech event, we naturally extend the vocabulary of the same encoder with phoneme tokens, realizing unified semantic modeling for text, timing, and phoneme features with a single text encoder.

With target distributions predefined by datasets constructed above, we introduce progressive diffusion training, fulfilling high-quality TTA synthesis at pre-training and gradually integrating fine-grained control signals at continual learning stages. At the first stage, we pre-train a diffusion transformer (DiT) in the latent space directly compressed from the waveform space, solely conditioned on text indication, achieving high-fidelity TTA at scale. At the second stage, we fine-tune the latent DiT on both text and timing conditions, enabling the model to precisely control the timing windows of each sound event. In controllable TTA generation, a common issue is the sacrifice of text-conditioned synthesis quality without fine-grained conditions (Wang et al., 2025d). Hence, in ControlAudio, we switch the condition between the text condition and the $\langle \text{text, timing} \rangle$ condition at the second stage, avoiding catastrophic forgetting in progressive training. At the final stage, given the audio generation prior learned in prior stages, we continually train the diffusion model by switching the condition among text, $\langle \text{text, timing} \rangle$, and $\langle \text{text, timing, phoneme} \rangle$, achieving high-fidelity audio synthesis conditioned on flexible indication.

In generation, diffusion models demonstrate a coarse-to-fine sampling nature. Along the entire trajectory, they generate large-scale features at the early stage and synthesize fine-grained details in the following steps, iteratively refining the generation results. In controllable TTA systems, condition signals show diverse control granularity as well. Hence, for timing-controlled and intelligible audio generation, we design progressive guided sampling, where the timing condition first guides the sampling to indicate the timing windows as large-scale features and then the phoneme condition is introduced to indicate the speech content as small-scale features. In comparison with a fixed guidance signal, our method gradually emphasizes more fine-grained condition information, inherently aligned with the diffusion sampling process. Extensive experiments demonstrate that ControlAudio achieves state-of-the-art performance on controllable audio generation tasks, significantly outperforming existing methods in both objective and subjective evaluations of temporal accuracy and speech clarity.

2 Related Work

2.1 Controllable TTA Generation

Recent works have explored temporal control in TTA generation through two main paradigms. Training-based methods, such as MC-Diffusion (Guo et al., 2024), PicoAudio (Xie et al., 2024) and Audio ControlNet (Zhu et al., 2026a), rely on predefined event classes, which limits their flexibility for open-domain prompts. More recent approaches, including AudioComposer (Wang et al., 2025d), PicoAudio2 (Zheng et al., 2025) and DegDiT (Liu et al., 2026), adopt natural language as the control interface, but often face ambiguity when describing complex multi-event scenarios. Alternatively, training-free methods such as TG-Diff (Du et al., 2024) and FreeAudio (Jiang et al., 2025) enforce temporal alignment during inference, but typically incur high computational overhead and struggle in dense event settings. In parallel, generating intelligible speech remains a fundamental challenge. Most TTA models produce speech as indistinct vocalizations. While models like VoiceLDM (Lee et al., 2024b), VoiceDiT (Jung et al., 2025) and UmbraTTS (Glazer et al., 2025) achieve high-quality speech synthesis, they are designed as specialized TTS systems and lack unified control over general audio events. Moreover, prior work on controllable dialogue, such as CoVoMix2 (Zhang et al., 2025), primarily focuses on speech-only scenarios. To address these limitations, we propose a unified framework for timing-controlled generation of general audio events and intelligible multi-speaker dialogue.

2.2 Progressive Modeling

In recent cross-modal generation tasks, such as video or avatar generation conditioned on diverse control signals (Zhao et al., 2025; Zhu et al., 2026b; Lin et al., 2025; Hu et al., 2025), progressive modeling has proven effective in handling multi-condition video generation. However, its advantages have only been partially explored in TTA generation, such as in long-form narrative audio (Guo et al., 2025) and video-to-audio (Tian et al., 2025; Dai et al., 2026) generation, and have not yet been extended to controllable TTA generation, where precise timing control and intelligible speech remain critical yet largely unresolved challenges. To address this gap, ControlAudio leverages progressive modeling to enable stage-wise controllability and fine-grained regulation across different levels.

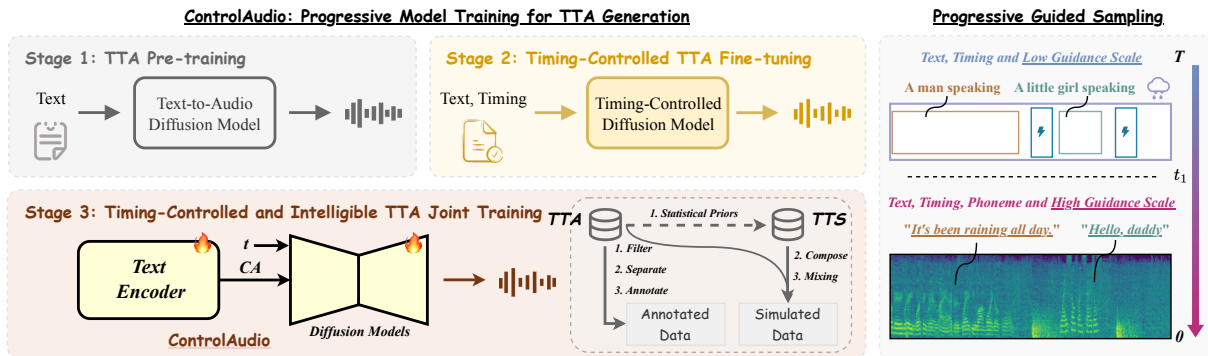


Figure 1: The end-to-end Progressive Diffusion Modeling of ControlAudio, which combines a progressive model training with a progressive guided sampling process for decoupled control of temporal structure and speech content.

3 Preliminary

3.1 Diffusion-based TTA Generation

Diffusion-based (Peebles and Xie, 2023; Li et al., 2024) TTA models are typically trained to learn a conditional reverse of a data-to-noise forward process (Ho et al., 2020), progressively removing noise from an initial random state conditioned on a text prompt over multiple diffusion steps. This framework consists of three main modules: 1) an audio variational autoencoder (VAE), responsible for transforming the audio sample into a compressed latent representation while ensuring the reconstruction quality; 2) a pretrained text encoder, which encodes a text prompt into conditioning embeddings; and 3) a latent diffusion model, which predicts the denoised audio latents conditioned on the text embeddings. In ControlAudio, we employ a DiT-based architecture to ensure scalability (Evans et al., 2025), conditioned on the text, timing, and phoneme embeddings to generate the latent audio representation directly compressed from the waveform, without cascaded decoding (Liu et al., 2023; Guo et al., 2024; Dai et al., 2025).

3.2 Classifier-Free Guidance

Classifier-Free Guidance (CFG) (Ho and Salimans, 2022; Wang et al., 2025b) emphasizes the guidance of a conditioning signal c during sampling. At each sampling step, CFG-guided diffusion models produce two predictions: a conditional estimation $\epsilon_{\theta}(x_t, c)$ and an unconditional estimation $\epsilon_{\theta}(x_t, \emptyset)$. Then the final prediction is obtained by extrapolating these two terms with a guidance scale $w > 1$:

$$\hat{\epsilon}_{\theta}(x_t, c) = \epsilon_{\theta}(x_t, \emptyset) + w \cdot (\epsilon_{\theta}(x_t, c) - \epsilon_{\theta}(x_t, \emptyset)). \quad (1)$$

Typically, a larger guidance scale w encourages stronger alignment with the condition, which may increase fidelity while sacrificing diversity.

4 ControlAudio

4.1 Motivation

As discussed above, current TTA generation quality has been advanced with latent diffusion models, while the quality of controllable generation, *e.g.*, precise timing control or intelligible speech control, is still limited. Although diverse innovations have been proposed, their synthesis quality at scale is still compromised by data scarcity. Moreover, previous research rarely achieves versatile TTA generation, namely, integrating additional fine-grained control signals while preserving high-fidelity audio generation solely conditioned on text.

In this work, we propose a progressive diffusion modeling design covering data construction and representation, model training, and guided sampling to tackle these difficulties, achieving text-guided, timing-indicated, and intelligible audio generation with a single diffusion model. Figure 1 illustrates our overall progressive strategy.

4.2 Dataset Construction

Data Scarcity. For TTA generation, we can collect various publicly available datasets, which comprise millions of weakly-labeled text-audio pairs (Appendix A.1), supporting high-quality synthesis at scale. However, these datasets typically contain only high-level textual descriptions, lacking the fine-grained annotations required for controllable synthesis (Wang et al., 2025a). Specifically, training timing-controlled and intelligible TTA generation requires datasets that combine speech with

general audio events under precise timing annotations. Yet, such datasets are rare: existing timing-annotated audio datasets are limited in scale and lack transcriptions for speech segments, while publicly available speech datasets do not have reliable temporal labels. To overcome this limitation, we first construct a multi-source dataset.

Annotated Data. Our data annotation pipeline begins with the AudioSet-SL (Hershey et al., 2021) dataset, chosen for its reliable temporal annotations while lacking corresponding speech transcripts. To create the ControlAudio dataset, we first select all clips containing "human speech" and then extract a clean speech track from each using a dual-demixing strategy inspired by MTV (Weng et al., 2025) that leverages both MVSEP (Solovyev) and Spleeter (Hennequin et al., 2020). The clean track is subsequently segmented into individual events using the original timestamps. Finally, each segmented event is transcribed using Gemini 2.5 Pro¹. Further details of the pipeline and prompt design are provided in Appendices A.3 and G. This transcription process enriches the dataset *i.e.*, expanding condition to $\langle \text{text}, \text{timing}, \text{phoneme} \rangle$ for fine-grained control. For example, a generic annotation like (man speaking, $\langle 3.00, 5.00 \rangle$) is transformed into a specific, content-rich event (man speaking: "It's been raining all day.", $\langle 3.00, 5.00 \rangle$).

Simulated Data. To further expand our dataset, we construct a large-scale simulated dataset guided by real-world data distribution. Specifically, we first analyze the AudioSet-SL dataset to derive statistical priors on speech activity patterns, with further details provided in Appendix A.4. These distributions guide our synthesis process, which proportionally simulates two main scenarios: single-speaker scenarios (*monologue*) are created by combining multiple utterances from the same speaker in LibriTTS-R, while multi-speaker scenarios (*dialogue*) are formed by sampling from different speakers. After composing the speech samples, we simulate a plausible temporal arrangement for the utterances. Finally, the composed speech is mixed with non-speech backgrounds from WavCaps (Mei et al., 2024) and VGG-Sound (Chen et al., 2020) at a signal-to-noise ratio sampled from a uniform 2 to 10 dB range (Jung et al., 2025). Through this simulation pipeline, we generate an additional 171,246 complex audio scenes, significantly expanding the scale and diversity of our training data.

¹<https://deepmind.google/models/gemini/pro/>

1. Text Prompt

Text: In the light rain with rumbling thunder, a man is speaking, then a little girl greets him.

2. Structured Prompt for Text and Timing Representation

Text and Timing: In the light rain with rumbling thunder, a man is speaking, then a little girl greets him.

```
@{Light rain & <0.00,10.00>}
@{Rumbling thunder & <5.00,5.75><8.00,8.75>}
@{A man speaking & <0.00,4.50>}
@{A little girl greets & <6.00,7.50>}
```

3. Structured Prompt for Phoneme Representation

Text, Timing and Phoneme: In the light rain with rumbling thunder, a man is speaking, then a little girl greets him.

```
@{Light rain & <0.00,10.00>}
@{Rumbling thunder & <5.00,5.75><8.00,8.75>}
@{A man speaking & <0.00,4.50><IH1><T><S><PAD><B><IH1>
<N><PAD><R><EYI><N><IH0><NG><PAD><AOI><L><PAD><D><EYI>}
@{A little girl greets & <6.00,7.50><HH><AH0><L><OWI><PAD><D>
<AEI><D><IYO>}
```

Speech Content: "It's been raining all day.", "Hello daddy!"

Figure 2: An illustrative example for structured prompt.

4.3 Unified Semantic Modeling

To address the challenge of encoding diverse condition information, including text, timing, and phoneme features, we propose a unified semantic modeling approach that handles them with a single text encoder in a progressive and coarse-to-fine manner. This approach avoids the complexity of multiple specialized modules (Lee et al., 2024b) by first establishing a robust structural representation, providing a simple yet effective solution for rendering fine-grained content in audio generation. **Structured Prompt for Text and Timing Representation.** The foundation of our approach is the Structured Prompt (y_s), a novel representation we design to explicitly and unambiguously define the composition of an acoustic scene. The prompt employs a standardized format using special tokens to delimit event descriptions and their precise start-and-end times, as illustrated in Figure 2. We propose this format to overcome the critical limitations of using free-form natural language for control. Natural language is often ambiguous; for instance, a prompt like "an alarm sounds from low to high from 1 second to 9 seconds" creates confusion, as a model must disentangle whether "from...to" refers to a change in pitch or a temporal boundary. Moreover, natural language descriptions become verbose and difficult to parse as scene complexity increases. In contrast, our structured format provides a concise, scalable, and machine-readable representation, forming a robust foundation for generating complex, temporally-aligned audio.

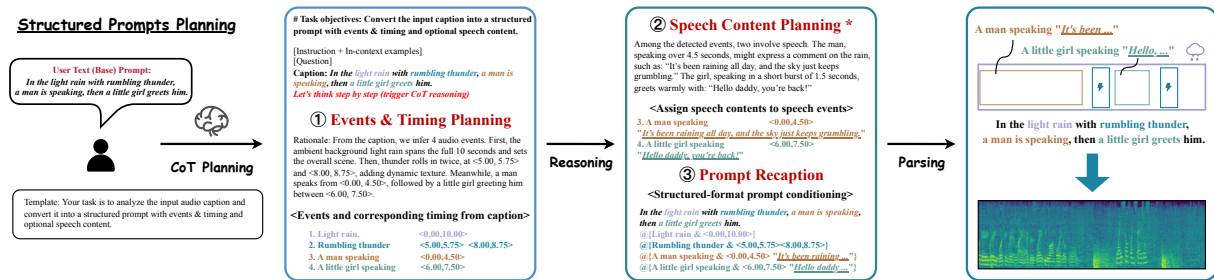


Figure 3: Overview of our CoT-based LLM planning pipeline. Given a user-provided free-form caption, the LLM performs multi-step reasoning to extract audio events with their temporal spans, infer speech content when applicable, and generate a structured prompt that encodes both timing and content for controllable audio generation.

Structured Prompt for Phoneme Representation.

Our approach to synthesizing intelligible speech is built upon the temporal structure provided by the prompt. A key insight is that the explicit timing windows (<start,end>) assigned to each speech event naturally define the total duration of the utterance. This design is consistent with recent TTS studies (Anastassiou et al., 2024; Lee et al., 2024a; Chen et al., 2025), which highlight the importance of duration information as a useful constraint for guiding speech generation, either through explicit modeling or by conditioning on the overall utterance duration. Building on this insight, our framework further organizes such temporal information into structured, event-level timing windows, enabling more fine-grained and interpretable control over the generation process.

This simplification makes it natural and highly efficient to use the same, single text encoder to progressively model both the coarse-grained temporal structure and the fine-grained speech content. We therefore represent the speech content at the phoneme level (e.g., "hello" → [HH, AH0, L, OW1]). Phonemes provide a more direct, pronunciation-aware signal than words, reducing ambiguity and improving the acoustic consistency of the generated speech. By augmenting our single encoder’s vocabulary with these phoneme tokens, it learns to render the precise phonetic sequence within the specified temporal boundaries, naturally inheriting the ability to handle speech duration.

Chain-of-Thought (CoT) LLM Planning. Given a user-provided free-form description, we employ a CoT-based LLM to convert it into the structured prompt format. The model decomposes the input into temporally-aligned audio events, infers speech content when applicable, and organizes them into a unified representation with explicit timing and phoneme-level information. This process resolves

ambiguities in natural language and ensures consistent alignment between semantic content and temporal structure, producing reliable control signals for controllable audio generation.

4.4 Progressive Model Training

To train our model for multi-condition audio generation, we adopt a progressive three-stage training strategy. This approach allows the model to acquire fine-grained control capability incrementally, where each new stage builds upon and refines the skills learned previously, ensuring a stable and efficient learning process. At each stage, the model is optimized using the conditional diffusion objective (Ho et al., 2020), where a network is trained to predict the noise ϵ added on the clean audio latents:

$$\mathcal{L} = \mathbb{E}_{x,c,\epsilon \sim \mathcal{N}(0,1),t} [\|\epsilon - \epsilon_{\theta}(z_t, t, \tau_{\theta}(c))\|_2^2], \quad (2)$$

where z_t is the noisy latent at timestep t , ϵ_{θ} is the denoising DiT, c is the condition signal, and τ_{θ} is the text encoder. The core of our progressive strategy lies in how the conditioning signal c is structured and utilized across the training stages.

Stage 1: TTA Pre-training. We first pre-train a DiT (Evans et al., 2025) on large-scale text-audio datasets to learn a robust, general mapping from textual descriptions to audio latent representation, ensuring *high-fidelity text-guided audio generation*.

Stage 2: Timing-Controlled TTA Fine-tuning. The pre-trained model is then fine-tuned on our dataset of precisely timing-annotated audio, while preserving the training on text condition without timing. This stage specifically optimizes the model to interpret the structured prompt containing both text and timing information, achieving *text-guided and timing-controlled audio generation*.

Stage 3: Timing-Controlled and Intelligible TTA Joint Training. At the final stage, we unfreeze the

text encoder to enable joint optimization for both timing control and speech intelligibility. The model is then trained on our full multi-source dataset, which is a comprehensive mixture of our timing-annotated real-world audio and the large-scale simulated data. This final training phase optimizes the model to jointly generate timing-controlled audio and speech samples in a coherent and realistic manner, addressing *text-guided*, *timing-controlled*, and *intelligible audio generation*.

Overall, our progressive model training incrementally acquires finer-grained capabilities while building upon the foundational skills from previous stages. Notably, we find that the joint optimization at Stage 3 not only unlocks speech intelligibility but also further enhances the model’s previously learned temporal precision. We attribute these significant improvements to two key factors. The first is the introduction of time-annotated speech data, which provides a richer, more targeted signal for learning the alignment between linguistic content and temporal boundaries. The second is the fine-tuning of the text encoder, which allows it to be jointly optimized with the diffusion backbone; this synergistic training enables both the conditioning (text encoder) and generation (DiT) components to co-adapt to the complex, multi-objective task. This effective co-adaptation is achieved within a simple yet effective framework, where a single text encoder is responsible for processing all conditioning signals: text, timing, and phoneme features.

4.5 Progressive Guided Sampling

To optimize our capability to handle both timing and more fine-grained phonetic content, we propose a Progressive Guided Sampling strategy. This approach divides the reverse diffusion process into two phases based on a threshold timestep t_1 , modulating the conditioning prompt and guidance scale accordingly. Specifically, in the initial sampling phase ($t \in [1.0, t_1]$), we guide the model with a simplified version of our structured prompt that excludes phonetic content c_1 , using a low guidance scale (w_{low}). This encourages the model to first establish a plausible temporal structure for all audio events:

$$p_\theta(z_{t_1:T-1}|z_T, c_1) = \prod_{t=t_1+1}^T p_\theta(z_{t-1}|z_t, c_1). \quad (3)$$

For the remainder of the sampling process ($t \in$

$(t_1, 0.0]$), we switch to the complete, phoneme-inclusive structured prompt c_2 and a higher guidance scale (w_{high}).

$$p_\theta(z_{0:t_1-1}|z_{t_1}, c_2) = \prod_{t=1}^{t_1} p_\theta(z_{t-1}|z_t, c_2). \quad (4)$$

This second phase strictly enforces adherence to the phonetic sequence, ensuring the synthesis of highly intelligible speech within the established structure. This coarse-to-fine strategy improves temporal accuracy and speech clarity by decoupling event placement and content rendering.

5 Experiments

5.1 Experiment Setting

Implementation Details. Our model follows a latent diffusion framework with a DiT backbone, where audio is generated in a compressed latent space learned by a DAC-VAE (Evans et al., 2024a). The autoencoder operates at a 16kHz sampling rate with a latent temporal resolution of 25Hz. The diffusion model adopts a DiT architecture following Stable Audio (Evans et al., 2024b, 2025), and generates 10-second audio segments in the latent space. It is conditioned via cross-attention on a pretrained Flan-T5 (Chung et al., 2024) large text encoder, where text, timing, and phoneme information are unified into a structured prompt and encoded within a shared embedding space. We first pretrain a TTA diffusion model for 1M steps, followed by two additional stages of optimization: 0.5M steps for timing-controlled generation and 0.5M steps with the text encoder unfrozen for joint training. Each stage is initialized from the checkpoint of the previous stage, forming a continuous multi-stage training pipeline. All stages are trained on 8 NVIDIA A800 GPUs with a total batch size of 128. More details on the model architecture and training are provided in the Appendix B. Pseudocode for both training and inference procedures is provided in Appendix D.

Evaluation Datasets. To objectively evaluate our method, we utilize several established datasets, each targeting a specific capability. For timing-controllable generation, we use the publicly available test split from AudioCondition (Guo et al., 2024), whose fine-grained temporal annotations are ideal for this task. For intelligible speech generation, we use the AC-Filtered (Lee et al., 2024b) dataset. To ensure comparability with prior methods, we rewrite the prompts accordingly, with

Table 1: Objective and subjective evaluation results on the AudioCondition test set. For each metric, the best result is bold and the second-best is underlined. * denotes models trained by ourselves. ‡ denotes models evaluated under a different SED model. ControlAudio full denotes evaluation on prompts covering all event classes in the test set.

Method	Temporal (Obj.)		Generation (Obj.)			Subjective		Efficiency
	Eb↑	At↑	FAD↓	KL↓	CLAP↑	Temporal↑	OVL↑	RTF↓
Ground Truth	43.37	67.53	-	-	0.377	4.52	4.48	-
AudioLDM Large	6.79	35.66	3.95	2.46	0.260	1.84	2.40	1.141
AudioLDM 2 Large	7.75	42.41	3.07	1.92	0.279	-	-	1.496
AudioLDM 2 Full Large	6.93	20.47	3.68	2.15	0.283	-	-	1.496
Tango	1.60	26.51	2.82	1.93	0.245	1.68	2.58	1.207
Stable Audio *	11.28	51.67	1.93	1.75	0.318	1.94	<u>3.44</u>	<u>0.821</u>
CCTA	14.57	18.27	-	-	-	-	-	1.207
MC-Diffusion	29.07	47.11	-	-	-	-	-	-
Tango + LControl	21.46	55.15	-	-	-	-	-	1.207
AudioComposer-Small	43.51	60.83	4.92	2.00	0.261	3.12	2.52	0.721
AudioComposer-Large	44.40	63.30	-	-	-	-	-	-
TG-Diff ‡	26.70	60.06	2.66	-	0.244	-	-	1.207
FreeAudio	44.34	68.50	<u>1.92</u>	<u>1.73</u>	0.321	-	-	1.166
ControlAudio	55.58	79.52	2.61	1.85	<u>0.325</u>	4.17	3.41	<u>0.821</u>
ControlAudio full	<u>49.85</u>	<u>71.55</u>	1.47	1.30	0.356	<u>3.96</u>	3.75	<u>0.821</u>

examples provided in Appendix F.2. For evaluating general TTA performance, we report results on the AudioCaps test set (Kim et al., 2019). In addition, we include the LibriTTS-R and LibriSpeech (Panayotov et al., 2015) *test-clean* splits for specific ablation studies.

Evaluation Metrics. We conduct a comprehensive evaluation covering three key aspects: temporal control, audio quality, and speech intelligibility. For temporal control, we follow prior work (Guo et al., 2024; Wang et al., 2025d) and report two metrics computed by a sound event detection (SED) system (Mesaros et al., 2016): event-based measures (Eb) and clip-level macro F1 score (At). For audio quality, we employ a suite of standard metrics, including Fréchet Audio Distance (FAD), Kullback–Leibler (KL) divergence, Fréchet Distance (FD), Inception Score (IS) (Liu et al., 2023) and CLAP (Wu et al., 2023). For speech intelligibility, we conduct both objective and subjective tests. Objectively, we measure the Word Error Rate (WER) by transcribing generated speech with the Whisper *Large-v3* model (Radford et al., 2023). Subjectively, we conduct Mean Opinion Score (MOS) tests where 20 participants rate three aspects on a five-point scale: Speech Intelligibility, Overall Quality (OVL), and Relevance to the prompt (REL). Further details are provided in the Appendix C.

5.2 Main Results

Timing-Controlled Audio Generation. We compare ControlAudio with several state-of-the-

art TTA models, including AudioLDM (Liu et al., 2023), AudioLDM 2 (Liu et al., 2024a), Tango (Ghosal et al., 2023), and our in-house version of Stable Audio (Evans et al., 2024a,b, 2025). We further include methods with explicit temporal conditioning, such as MC-Diffusion (Guo et al., 2024) and AudioComposer (Wang et al., 2025d), as well as training-free baselines TG-Diff (Du et al., 2024) and FreeAudio (Jiang et al., 2025), covering a broad range of paradigms for controllable TTA generation. TG-Diff reports both timing and audio quality metrics under a training-free framework, but relies on a different sound event detection model (Turpault et al., 2019) from other baselines. CCTA is a variant of MC-Diffusion that uses only control conditions without textual input, while Tango + LControl extends Tango with language-based temporal control. For AudioComposer, we use its publicly available Small variant for evaluation. We evaluate all methods from three perspectives: temporal alignment, audio quality, and efficiency. For efficiency, we measure the real-time factor (RTF) (Liu et al., 2024b,c) on a single NVIDIA A800 GPU. As shown in Table 1, ControlAudio achieves competitive or superior temporal alignment compared to existing methods, while consistently improving audio quality in both objective and subjective metrics. A detailed category-wise breakdown on AudioCondition is provided in Appendix E.1, reporting Eb and At metrics across different event categories. Importantly, these im-

Table 2: Objective and subjective evaluation results on the AC-Filtered.

Method	Objective						Subjective		
	FAD↓	KL↓	FD↓	IS↑	CLAP↑	WER↓	Intelligible↑	OVL↑	REL↑
Ground Truth	-	-	-	-	0.523	17.47	4.16	4.45	4.50
AudioLDM 2 Speech	23.55	3.58	102.84	1.52	0.078	32.74	2.85	1.92	1.60
VoiceLDM-S	4.46	1.52	47.08	<u>3.40</u>	<u>0.479</u>	43.21	2.62	2.55	2.51
VoiceLDM-M	5.90	1.43	<u>46.40</u>	3.16	0.458	8.84	<u>4.18</u>	<u>3.64</u>	<u>3.47</u>
VoiceDiT	<u>4.60</u>	-	-	-	0.220	<u>7.09</u>	-	-	-
ControlAudio	3.52	<u>1.45</u>	32.55	4.43	0.513	6.84	4.31	4.15	3.82

improvements are achieved without introducing additional inference overhead, demonstrating a favorable trade-off between controllability, quality, and efficiency.

Intelligible Audio Generation. We further assess the ability of ControlAudio to generate intelligible speech on the AC-Filtered, comparing it with speech-oriented baselines including AudioLDM 2 Speech, VoiceLDM-S, VoiceLDM-M, and VoiceDiT. To evaluate these baselines lacking native timing support, we first use an LLM to predict a plausible time window from the caption, with further details in the Appendix F.2. For this comparison, we use the publicly available checkpoints of VoiceLDM, while directly reporting the results presented in the original VoiceDiT paper. As shown in Table 2, ControlAudio achieves lower WER, superior audio quality, and improved text-audio semantic alignment compared to all baselines. Subjective evaluations also indicate improvements in speech intelligibility, overall audio quality, and text relevance, demonstrating that ControlAudio can generate clearer and more faithful speech segments while preserving general audio fidelity.

Text-to-Audio Generation. To verify that introducing timing and speech content control does not compromise general TTA generation capabilities, we evaluate ControlAudio on the AudioCaps test set under standard natural language captions, comparing against both text-to-audio models and other controllable audio generation baselines. Unlike prior controllable generation approaches that often sacrifice audio quality for control precision, ControlAudio maintains high generative performance while providing fine-grained controllability. As shown in Table 3, ControlAudio achieves competitive or superior results across multiple audio quality metrics compared to state-of-the-art baselines. These findings demonstrate that our structured prompt conditioning and vocabulary exten-

Table 3: Objective and subjective evaluation results on the AudioCaps test set.

Method	FAD↓	KL↓	FD↓	IS↑	CLAP↑
Ground Truth	-	-	-	-	0.525
AudioGen	1.82	1.69	-	-	-
AudioLDM	4.96	2.17	29.29	8.13	0.373
AudioLDM 2	2.12	1.54	33.18	8.29	0.281
Tango	1.73	<u>1.27</u>	24.42	7.70	0.315
Tango 2	2.63	1.12	20.66	9.09	0.375
Stable Audio *	1.52	1.51	<u>18.30</u>	<u>13.79</u>	0.538
AudioComposer-S	3.63	1.76	27.57	-	-
AudioComposer-L	2.52	1.39	19.25	-	-
VoiceLDM-S	13.83	3.36	63.42	4.56	0.217
VoiceLDM-M	9.70	2.81	55.80	4.60	0.272
VoiceDiT	3.55	1.87	-	-	0.450
ControlAudio	<u>1.56</u>	1.31	14.20	14.49	<u>0.535</u>

sion can be seamlessly integrated into a T2A system, enabling precise timing and intelligible speech control without degrading semantic alignment or acoustic fidelity.

5.3 Ablation Study

Ablation of Prompt Design. To isolate and evaluate the effectiveness of our structured prompt design, we conduct a targeted ablation study. For this analysis, we compare a baseline model trained with conventional natural language descriptions against our model trained with structured prompts. Crucially, both models are trained only up to Stage 2 of our progressive curriculum, the phase dedicated specifically to learning timing control. This controlled setting allows us to fairly assess the impact of the prompt format itself. As shown in Table 4, the model trained with structured prompts consistently achieves superior temporal alignment and overall audio quality on the AudioCondition test set. The results suggest that the structured format provides a clearer, unambiguous mapping between events and their time spans, an advantage that becomes particularly pronounced in complex scenes where verbose natural language descriptions can degrade timing accuracy.

Table 4: Comparison of prompt formats on the Audio-Condition test set. We compare Natural Language (NL) with our Structured Prompt (SP).

Format	Eb \uparrow	At \uparrow	FAD \downarrow	KL \downarrow	CLAP \uparrow
NL	46.23	65.36	4.11	2.25	0.245
SP	51.62	70.81	3.61	2.05	0.293
NL full	40.79	61.06	1.03	1.36	0.376
SP full	43.76	64.82	0.92	1.27	0.419

Table 5: Comparison of vocabulary extension strategies for intelligible speech synthesis on LibriTTS-R and LibriSpeech test sets.

Token Type	LibriTTS-R		LibriSpeech	
	WER \downarrow	UT-M \uparrow	WER \downarrow	UT-M \uparrow
Ground Truth	3.75	4.17	2.15	4.06
VoiceLDM-S	36.65	2.59	38.61	2.76
VoiceLDM-M	4.98	2.83	9.76	2.77
Word	6.96	4.12	6.44	4.14
BPE	7.53	4.15	5.04	4.20
Phoneme	4.00	4.18	3.62	4.22

Ablation of Vocabulary Granularity. To determine the optimal vocabulary granularity for intelligible speech, we conduct an ablation study on the LibriTTS-R and LibriSpeech test-clean datasets. We first compare three variants of our model, differentiated by their vocabulary: word-level, sub-word (BPE), and phoneme-level. For broader context, we also report results from the strong VoiceLDM baselines. Evaluation metrics include WER for intelligibility and UTMOS (Saeki et al., 2022)(UT-M) for speech naturalness. For WER calculation on LibriSpeech, we adopt a HuBERT-based ASR model (Hsu et al., 2021), following prior works (Shen et al., 2023). As shown in Table 5, the phoneme-level model consistently and significantly outperforms the other granularities, achieving the lowest WER and highest UTMOS scores. These findings confirm that phonemes provide a more direct representation of spoken content, facilitating a tighter alignment between the prompt and the acoustic output, which ultimately translates into substantially clearer and more intelligible speech.

Analysis of Sampling Strategy. We conduct an analysis to validate our progressive sampling strategy. This coarse-to-fine approach first uses a low guidance scale (w_{low}) with a simplified, content-free prompt to establish the temporal structure. It then transitions to a high scale (w_{high}) with the full, phoneme-inclusive prompt to render intelligible speech. As visualized in Figure 4, our analysis

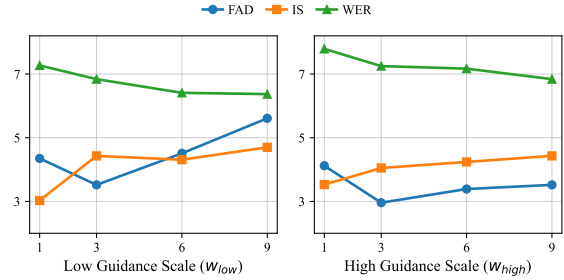


Figure 4: Analysis of Progressive Sampling parameters (w_{low} , w_{high}). This study reveals a clear trade-off between audio quality and speech intelligibility.

of varying w_{low} and w_{high} reveals a clear trade-off: a low initial scale is crucial for overall audio quality, while a high subsequent scale is essential for speech intelligibility. This study empirically identifies the optimal configuration as ($w_{low} = 3, w_{high} = 9$), confirming the effectiveness of our approach. For a total of $T = 100$ sampling steps, we set the transition timestep to $t_1 = 88$. We also examine the sensitivity of t_1 : a smaller t_1 (longer low-guidance phase) improves audio quality, while a larger t_1 reduces the steps for phoneme refinement and harms intelligibility. The performance remains stable across a reasonable range of t_1 . Detailed sensitivity results of t_1 are provided in Appendix E.2.

6 Conclusion

In this work, we introduced ControlAudio, which recasts controllable TTA generation as a multi-task learning problem solved via a progressive diffusion modeling strategy. This progressive approach is applied across data construction, model training, and inference, enabling our model to incrementally master fine-grained control from text, timing, and phoneme conditions. Extensive experiments demonstrate that ControlAudio achieves state-of-the-art performance in both temporal accuracy and speech clarity. Our work’s potential for misuse in creating deceptive content or voice impersonations underscores the urgent need for robust detection methods and responsible AI governance.

7 Acknowledgments

This work is supported by the Fundamental and Interdisciplinary Disciplines Breakthrough Plan of the Ministry of Education of China (No. JYB2025XDXM101), and the National Natural Science Foundation of China (62550004, U24A20342, U25B6003, 92570001).

Limitations

Despite its promising results, our work has several limitations. First, while ControlAudio pioneers the generation of intelligible speech within a timing-controlled TTA framework, its control is primarily limited to the speech content. The framework currently lacks explicit mechanisms to manipulate crucial stylistic attributes such as emotion, prosody, or speaker identity. Second, a fundamental tension between generating high-quality general audio versus intelligible speech persists. Although our model unifies these tasks, we observe a potential trade-off where heavily optimizing for one modality can slightly impact the fidelity of the other in complex, co-occurring scenes. Finally, the performance of our model is inherently constrained by the availability of large-scale, richly annotated audio-speech datasets, which remain scarce. Our reliance on a combination of existing annotated data and simulated data, while effective, suggests that performance could be further enhanced with the advent of more comprehensive and higher-quality training corpora in the future.

References

- Philip Anastassiou, Jiawei Chen, Jitong Chen, Yuanzhe Chen, Zhuo Chen, Ziyi Chen, Jian Cong, Lelai Deng, Chuang Ding, Lu Gao, and 1 others. 2024. Seed-tts: A family of high-quality versatile speech generation models. *arXiv preprint arXiv:2406.02430*.
- Honglie Chen, Weidi Xie, Andrea Vedaldi, and Andrew Zisserman. 2020. Vggsound: A large-scale audio-visual dataset. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 721–725. IEEE.
- Yushen Chen, Zhikang Niu, Ziyang Ma, Keqi Deng, Chunhui Wang, JianZhao JianZhao, Kai Yu, and Xie Chen. 2025. F5-tts: A fairytaler that fakes fluent and faithful speech with flow matching. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6255–6271.
- Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, and 1 others. 2024. Scaling instruction-finetuned language models. *Journal of Machine Learning Research*, 25(70):1–53.
- Yusheng Dai, Zehua Chen, Yuxuan Jiang, Baolong Gao, Qiuhong Ke, Jun Zhu, and Jianfei Cai. 2026. Omni2sound: Towards unified video-text-to-audio generation. *arXiv preprint arXiv:2601.02731*.
- Yusheng Dai, Chenxi Wang, Chang Li, Chen Wang, Jun Du, Kewei Li, Ruoyu Wang, Jiefeng Ma, Lei Sun, and Jianqing Gao. 2025. Latent swap joint diffusion for long-form audio generation. *arXiv preprint arXiv:2502.05130*.
- Tianjiao Du, Jun Chen, Jiasheng Lu, Qinmei Xu, Huan Liao, Yupeng Chen, and Zhiyong Wu. 2024. Controllable text-to-audio generation with training-free temporal guidance diffusion. In *2024 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE.
- Zach Evans, CJ Carr, Josiah Taylor, Scott H Hawley, and Jordi Pons. 2024a. Fast timing-conditioned latent audio diffusion. In *Forty-first International Conference on Machine Learning*.
- Zach Evans, Julian D Parker, CJ Carr, Zack Zukowski, Josiah Taylor, and Jordi Pons. 2024b. Long-form music generation with latent diffusion. *arXiv preprint arXiv:2404.10301*.
- Zach Evans, Julian D Parker, CJ Carr, Zack Zukowski, Josiah Taylor, and Jordi Pons. 2025. Stable audio open. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.
- 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. IEEE.
- Deepanway Ghosal, Navonil Majumder, Ambuj Mehrih, and Soujanya Poria. 2023. Text-to-audio generation using instruction guided latent diffusion model. In *Proceedings of the 31st ACM International Conference on Multimedia*, pages 3590–3598.
- Neta Glazer, Aviv Navon, Yael Segal, Aviv Shamsian, Hilit Segev, Asaf Buchnick, Menachem Pirchi, Gil Hetz, and Joseph Keshet. 2025. Umbratts: Adapting text-to-speech to environmental contexts with flow matching. *arXiv preprint arXiv:2506.09874*.
- Yuxin Guo, Teng Wang, Yuying Ge, Shijie Ma, Yixiao Ge, Wei Zou, and Ying Shan. 2025. Audiostory: Generating long-form narrative audio with large language models. *arXiv preprint arXiv:2508.20088*.
- Zhifang Guo, Jianguo Mao, Rui Tao, Long Yan, Kazushige Ouchi, Hong Liu, and Xiangdong Wang. 2024. Audio generation with multiple conditional diffusion model. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 18153–18161.
- Romain Hennequin, Anis Khlif, Felix Voituret, and Manuel Moussallam. 2020. Spleeter: a fast and efficient music source separation tool with pre-trained models. *Journal of Open Source Software*, 5(50):2154.

- Shawn Hershey, Daniel PW Ellis, Eduardo Fonseca, Aren Jansen, Caroline Liu, R Channing Moore, and Manoj Plakal. 2021. The benefit of temporally-strong labels in audio event classification. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 366–370. IEEE.
- Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems*, 33:6840–6851.
- Jonathan Ho and Tim Salimans. 2022. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598*.
- Wei-Ning Hsu, Benjamin Bolte, Yao-Hung Hubert Tsai, Kushal Lakhotia, Ruslan Salakhutdinov, and Abdelrahman Mohamed. 2021. Hubert: Self-supervised speech representation learning by masked prediction of hidden units. *IEEE/ACM transactions on audio, speech, and language processing*, 29:3451–3460.
- Teng Hu, Zhentao Yu, Zhengguang Zhou, Sen Liang, Yuan Zhou, Qin Lin, and Qinglin Lu. 2025. Hunyuancustom: A multimodal-driven architecture for customized video generation. *arXiv preprint arXiv:2505.04512*.
- 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. In *International Conference on Machine Learning*, pages 13916–13932. PMLR.
- Yuxuan Jiang, Zehua Chen, Zeqian Ju, Chang Li, Weibei Dou, and Jun Zhu. 2025. Freeaudio: Training-free timing planning for controllable long-form text-to-audio generation. *arXiv preprint arXiv:2507.08557*.
- Jaemin Jung, Junseok Ahn, Chaeyoung Jung, Tan Dat Nguyen, Youngjoon Jang, and Joon Son Chung. 2025. Voicedit: Dual-condition diffusion transformer for environment-aware speech synthesis. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.
- 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.
- Yuma Koizumi, Heiga Zen, Shigeki Karita, Yifan Ding, Kohei Yatabe, Nobuyuki Morioka, Michiel Bacchiani, Yu Zhang, Wei Han, and Ankur Bapna. 2023. Libritts-r: A restored multi-speaker text-to-speech corpus. *arXiv preprint arXiv:2305.18802*.
- Keon Lee, Dong Won Kim, Jaehyeon Kim, and Jaewoong Cho. 2024a. Ditto-tts: Efficient and scalable zero-shot text-to-speech with diffusion transformer. *arXiv e-prints*, pages arXiv–2406.
- Yeonghyeon Lee, Inmo Yeon, Juhan Nam, and Joon Son Chung. 2024b. Voiceldm: Text-to-speech with environmental context. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 12566–12571. IEEE.
- Chang Li, Ruoyu Wang, Lijuan Liu, Jun Du, Yixuan Sun, Zilu Guo, Zhenrong Zhang, Yuan Jiang, Jianqing Gao, and Feng Ma. 2024. Qa-mdt: Quality-aware masked diffusion transformer for enhanced music generation. *arXiv preprint arXiv:2405.15863*.
- Gaojie Lin, Jianwen Jiang, Jiaqi Yang, Zerong Zheng, and Chao Liang. 2025. Omnihuman-1: Rethinking the scaling-up of one-stage conditioned human animation models. *arXiv preprint arXiv:2502.01061*.
- 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. *arXiv preprint arXiv:2301.12503*.
- Haohe Liu, Yi Yuan, Xubo Liu, Xinhao Mei, Qiuqiang Kong, Qiao Tian, Yuping Wang, Wenwu Wang, Yuxuan Wang, and Mark D Plumbley. 2024a. Audioldm 2: Learning holistic audio generation with self-supervised pretraining. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.
- Huadai Liu, Rongjie Huang, Yang Liu, Hengyuan Cao, Jialei Wang, Xize Cheng, Siqi Zheng, and Zhou Zhao. 2024b. Audioldm: Text-to-audio generation with latent consistency models. *arXiv preprint arXiv:2406.00356*.
- Huadai Liu, Jialei Wang, Rongjie Huang, Yang Liu, Heng Lu, Zhou Zhao, and Wei Xue. 2024c. Flashaudio: Rectified flows for fast and high-fidelity text-to-audio generation. *arXiv preprint arXiv:2410.12266*.
- Yisu Liu, Chenxing Li, Wanqian Zhang, Wenfu Wang, Meng Yu, RuiBo Fu, Zheng Lin, Weiping Wang, and Dong Yu. 2026. Degdit: Controllable audio generation with dynamic event graph guided diffusion transformer. *IEEE Transactions on Audio, Speech and Language Processing*.
- Xinhao Mei, Chutong Meng, Haohe Liu, Qiuqiang Kong, Tom Ko, Chengqi Zhao, Mark D Plumbley, Yuexian Zou, and Wenwu Wang. 2024. Wavcaps: A chatgpt-assisted weakly-labelled audio captioning dataset for audio-language multimodal research. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*.
- Annamaria Mesaros, Toni Heittola, and Tuomas Virtanen. 2016. Metrics for polyphonic sound event detection. *Applied Sciences*, 6(6):162.

- Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: an asr corpus based on public domain audio books. In *2015 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pages 5206–5210. IEEE.
- William Peebles and Saining Xie. 2023. Scalable diffusion models with transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 4195–4205.
- Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2023. Robust speech recognition via large-scale weak supervision. In *International conference on machine learning*, pages 28492–28518. PMLR.
- Takaaki Saeki, Detai Xin, Wataru Nakata, Tomoki Koriyama, Shinnosuke Takamichi, and Hiroshi Saruwatari. 2022. Utmos: Utokyo-sarulab system for voicemos challenge 2022. *Interspeech 2022*.
- Kai Shen, Zeqian Ju, Xu Tan, Yanqing Liu, Yichong Leng, Lei He, Tao Qin, Sheng Zhao, and Jiang Bian. 2023. Naturalspeech 2: Latent diffusion models are natural and zero-shot speech and singing synthesizers. *arXiv preprint arXiv:2304.09116*.
- Roman Solovyev. Cinematic sound demixing. <https://github.com/ZFTurbo/MVSEP-CDX23-Cinematic-Sound-Demixing>.
- Zeyue Tian, Yizhu Jin, Zhaoyang Liu, Ruibin Yuan, Xu Tan, Qifeng Chen, Wei Xue, and Yike Guo. 2025. Audiox: Diffusion transformer for anything-to-audio generation. *arXiv preprint arXiv:2503.10522*.
- Nicolas Turpault, Romain Serizel, Ankit Parag Shah, and Justin Salamon. 2019. Sound event detection in domestic environments with weakly labeled data and soundscape synthesis. In *Workshop on Detection and Classification of Acoustic Scenes and Events*.
- Chenxi Wang, Yusheng Dai, Lei Sun, Jun Du, and Jianqing Gao. 2025a. Audioatlas: A comprehensive and balanced benchmark towards movie-oriented text-to-audio generation. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pages 13266–13272.
- Junyou Wang, Zehua Chen, Binjie Yuan, Kaiwen Zheng, Chang Li, Yuxuan Jiang, and Jun Zhu. 2025b. Audiomog: Guiding audio generation with mixture-of-guidance. *arXiv preprint arXiv:2509.23727*.
- Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, and 1 others. 2024. Qwen2-vl: Enhancing vision-language model’s perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*.
- Yaoting Wang, Shengqiong Wu, Yuecheng Zhang, Shuicheng Yan, Ziwei Liu, Jiebo Luo, and Hao Fei. 2025c. Multimodal chain-of-thought reasoning: A comprehensive survey. *arXiv preprint arXiv:2503.12605*.
- Yuanyuan Wang, Hangting Chen, Dongchao Yang, Zhiyong Wu, and Xixin Wu. 2025d. Audiocomposer: Towards fine-grained audio generation with natural language descriptions. In *ICASSP 2025-2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.
- Shuchen Weng, Haojie Zheng, Zheng Chang, Si Li, Boxin Shi, and Xinlong Wang. 2025. Audio-sync video generation with multi-stream temporal control. *arXiv preprint arXiv:2506.08003*.
- Yusong Wu, Ke Chen, Tianyu Zhang, Yuchen Hui, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. 2023. Large-scale contrastive language-audio pretraining with feature fusion and keyword-to-caption augmentation. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.
- Zeyu Xie, Xuenan Xu, Zhizheng Wu, and Mengyue Wu. 2024. Picoaudio: Enabling precise timestamp and frequency controllability of audio events in text-to-audio generation. *arXiv preprint arXiv:2407.02869*.
- Leying Zhang, Yao Qian, Xiaofei Wang, Manthan Thakker, Dongmei Wang, Jianwei Yu, Haibin Wu, Yuxuan Hu, Jinyu Li, Yanmin Qian, and 1 others. 2025. Covomix2: Advancing zero-shot dialogue generation with fully non-autoregressive flow matching. *arXiv preprint arXiv:2506.00885*.
- Min Zhao, Rongzhen Wang, Fan Bao, Chongxuan Li, and Jun Zhu. 2025. Controlvideo: conditional control for one-shot text-driven video editing and beyond. *Science China Information Sciences*, 68(3):132107.
- Zihao Zheng, Zeyu Xie, Xuenan Xu, Wen Wu, Chao Zhang, and Mengyue Wu. 2025. Picoaudio2: Temporal controllable text-to-audio generation with natural language description. *arXiv preprint arXiv:2509.00683*.
- Haina Zhu, Yao Xiao, Xiquan Li, Ziyang Ma, Jianwei Yu, Bowen Zhang, Mingqi Yang, and Xie Chen. 2026a. Audio controlnet for fine-grained audio generation and editing. *arXiv preprint arXiv:2602.04680*.
- Hongzhou Zhu, Min Zhao, Guande He, Hang Su, Chongxuan Li, and Jun Zhu. 2026b. Causal forcing: Autoregressive diffusion distillation done right for high-quality real-time interactive video generation. *arXiv preprint arXiv:2602.02214*.

A Training Datasets

A.1 Pretraining Datasets and Preprocessing

Table 6 provides a comprehensive summary of all corpora used for pretraining our TTA backbone. To learn a robust mapping between text and audio, we aggregate a diverse mixture of large-scale, publicly available datasets. This includes datasets with descriptive captions, such as the large-scale WavCaps (Mei et al., 2024) and the widely-used AudioCaps (Kim et al., 2019), as well as corpora with high-level event labels, like the massive AudioSet (Gemmeke et al., 2017). This rich combination of data sources, spanning both detailed descriptions and a wide vocabulary of sound classes, allows the model to learn robust and versatile semantic representations.

All audio samples from these sources undergo a standardized preprocessing pipeline. First, all audio is resampled to 16kHz and converted to a mono-channel format. To accommodate the fixed-size input requirement of our diffusion model, all clips are processed into a uniform 10-second duration. Samples shorter than 10 seconds are right-padded with silence, while for samples longer than 10 seconds, a random 10-second segment is cropped.

Table 6: Details about audio-text datasets we use.

Dataset	Hours(h)	Number	Text
AudioCaps	109	44K	caption
WavCaps	7090	400K	caption
Clotho v2	152	7k	caption
AudioSet	5800	2M	label
FSD50k	108	51K	label
ESC-50	2.8	2K	label
VGG-Sound	550	210k	label
MTT	200	24K	caption
MSD	7333	880K	caption
FMA	900	11K	caption

A.2 Timing-Controlled Datasets

For the timing-control fine-tuning stage, our dataset is constructed based on AudioSet-Strong (Hershey et al., 2021), which contains 1.8M audio clips. This dataset is crucial as it provides dense, frame-level timestamps for 456 sound event classes. However, since AudioSet-Strong only provides categorical labels (e.g., "Dog"), not descriptive text, we generate richer captions for each timed event. Inspired by the methodology of WavCaps (Mei et al., 2024),

we employ a large language model (LLM) to create a unique textual description for each segmented audio event.

A critical difference in preprocessing this dataset is the handling of audio duration to preserve the integrity of the timestamps. Unlike in the pretraining phase, we do not apply random cropping. Instead, we consistently take the first 10 seconds of each audio clip and subsequently filter the event annotations, retaining only those whose timestamps fall within this 0-10s window. Clips shorter than 10 seconds are right-padded with silence. This deterministic process ensures that the temporal annotations in our final dataset remain perfectly aligned with the corresponding audio segments.

A.3 Annotated Data Pipeline

This section provides a detailed, step-by-step description of the pipeline used to create our annotated dataset of real-world speech events with both precise temporal boundaries and textual transcriptions. The process is as follows:

Initial Data Selection from AudioSet-SL. We begin with the AudioSet-SL dataset (Hershey et al., 2021), which contains strong, human-verified temporal annotations for a wide range of sound events. From the full dataset, we first identify and select all 10-second audio clips that contain at least one event labeled as "Human speech," "Speech," or any of their subcategories. This initial filtering yields a subset of 49,950 clips containing speech.

High-Quality Speech Track Extraction. To obtain high-quality and reliable speech stems, we employ a dual-demixing comparison strategy inspired by MTV (Weng et al., 2025). This strategy involves comparing the separation outputs from MVSEP (Solovyev) and Spleeter (Hennequin et al., 2020) to filter for quality and extract a clean speech signal for the subsequent processing steps.

Event-Level Segmentation. The extracted clean speech track is then segmented into individual, non-overlapping speech events. We use the original, human-annotated start and end timestamps provided by AudioSet-SL to perform this segmentation. Each resulting audio segment represents a single, continuous speech utterance from the original recording. This process yields a total of 173,831 individual speech segments, which are then prepared for transcription.

Transcription with Large Language Model. Each of the 173,831 clean, segmented speech events is then sent for transcription. We input the

audio segment into the Gemini 2.5 Pro model with a direct prompt to generate a precise textual transcription. To ensure the quality of the final annotations, we explicitly instruct the model to return an empty output if the spoken content in an audio segment is unintelligible or heavily obscured by noise. This step serves as a crucial quality filter.

This entire pipeline, from segmentation to filtered transcription, results in our final annotated dataset. From the initial pool of segments, a total of 152,070 high-quality, transcribed speech events are retained. Each event in this dataset is characterized by a precise start time, end time, and a verified textual transcription, providing an authentic and challenging data source for training our model on real-world, timed speech.

A.4 Simulated Data Pipeline

In addition to the annotated real-world data, we developed a pipeline to construct a large-scale simulated dataset. The goal of this pipeline is to generate realistic, complex audio scenes with precise timing and transcription information, guided by the statistical patterns observed in a real-world dataset. The process consists of two main stages: deriving statistical priors and the guided synthesis itself.

Deriving Statistical Priors from AudioSet-SL

To ensure our simulated data reflects real-world patterns of speech activity, we first perform a statistical analysis on the speech-containing clips within AudioSet-SL. We identify two key distributions:

- **Speaker Distribution:** We find that approximately 79.1% of clips (39,509 out of 49,950) feature a single speaker, while 20.9% feature multiple speakers. This ratio guides the proportion of monologue vs. dialogue scenarios in our simulation.
- **Utterance-per-Clip Distribution:** The empirical distribution is characterized by a prominent peak at a single utterance per clip ($n = 1$), which accounts for 32.20% of all single-speaker scenarios. For $n > 1$, the frequency of clips generally decreases as the number of utterances increases, exhibiting a long tail. We sample from this distribution to determine the number of utterances in our simulated monologues, with the maximum number of utterances per clip capped at 8 to focus on the most prevalent scenarios. The full distribution is provided in Table 7.

Guided Synthesis Pipeline. The synthesis process for each 10-second clip is as follows. First, we

Table 7: Empirical distribution of the number of utterances per 10-second single-speaker clip, analyzed from AudioSet-SL. This distribution guides the synthesis of our simulated monologue data.

Events (n)	Number	Percentage (%)
1	12,723	32.20
2	6,462	16.36
3	6,284	15.90
4	5,720	14.48
5	4,201	10.63
6	2,328	5.89
7	1,047	2.65
8	456	1.15
9	150	0.38
10	67	0.17
11	41	0.10
12	20	0.05
$n > 12$	10	0.04
Total	39,509	100.00

determine the scenario type by sampling from the speaker distribution (a 79.1% chance of a single-speaker monologue). Next, we source clean speech utterances with transcripts from the LibriTTS-R dataset (Koizumi et al., 2023). For a monologue, we sample a number of utterances (determined by the utterance-per-clip distribution, and capped at a maximum of 8) from a single speaker. For a dialogue, we sample utterances from 2 to 4 different speakers, ensuring that no single speaker contributes more than 4 utterances. We then simulate a plausible temporal arrangement for these utterances within the 10-second window. Finally, the composed speech-only track is mixed with a non-speech background audio clip randomly selected from a filtered subset of WavCaps (Mei et al., 2024) and VGG-Sound (Chen et al., 2020). The mixing is performed at a signal-to-noise ratio (SNR) randomly sampled from a uniform distribution between 2 and 10 dB.

B Model Configurations

This section details the architecture of the base model used for pretraining, before it is fine-tuned into ControlAudio. Our diffusion model is built upon the DiT (Diffusion Transformer) architecture within a latent diffusion modeling (LDM) paradigm. For pretraining, the model is conditioned on three input types: a natural language

prompt (prompt), the start time (seconds_start), and the total duration (seconds_total). All conditions are embedded into a 768-dimensional feature space. The prompt is encoded using a pretrained Flan-T5 large model, while seconds_start and seconds_total are treated as numerical inputs.

The diffusion network backbone is a DiT with 24 layers, 24 attention heads, and a model hidden dimension of 1536 (Evans et al., 2024b). The model utilizes both cross-attention for all conditional inputs and global conditioning for duration-related signals. The internal token dimension of the diffusion model is 64, with a conditional token dimension of 768 and a global condition embedding dimension of 1536.

B.1 Compression Networks

Our audio autoencoder is a variational autoencoder (VAE) based on the Describe Audio VAE (Evans et al., 2025) framework, operating at a 16kHz sampling rate. The model is trained from scratch on the audio portions of large-scale public datasets to learn a compact audio representation. The encoder is configured with a model dimension (d_model) of 128 and uses strides of [4, 4, 4, 10], resulting in an overall downsampling ratio of 640. The encoder maps the input waveform into a final 64-dimensional latent representation, which is then used by the decoder for reconstruction. The model’s input/output channels (io_channels) are set to 1 for mono audio. We use Snake activation throughout the network and omit the final tanh activation in the decoder.

B.2 Training Details

To improve convergence stability and generation quality, we adopt several common training strategies (Evans et al., 2025), with configurations specified in our training setup. We apply Exponential Moving Average (EMA) to the model parameters. For optimization, we use the AdamW optimizer with a learning rate of 5×10^{-5} , $(\beta_1, \beta_2) = (0.9, 0.999)$, and a weight decay of 1×10^{-3} .

Our learning rate schedule consists of two phases. For the first 99% of training iterations, the learning rate is held constant at its initial value, η_0 . For the final 1% of iterations, it then decays following the InverseLR formula:

$$\eta_t = \eta_0 \times \left(1 + \frac{t'}{\gamma}\right)^{-\text{power}}, \quad (5)$$

where t' is the step count within the decay phase,

$\gamma = 10^6$, and power = 0.5. This strategy allows for stable and rapid convergence in the main training phase, followed by a short period of fine-tuning with a decaying learning rate.

In this final stage, we initialize the model from the Stage 2 checkpoint and unfreeze the Flan-T5 text encoder, enabling joint optimization with the diffusion backbone. The optimization configurations are retained from the previous stages. This joint training is crucial as it allows the text encoder to adapt its representations to the composite nature of our prompt, which includes the structured format, special tokens for timing, and the extended phoneme-level vocabulary for speech. As a result, the model learns a unified representation that maps diverse inputs, such as semantic descriptions, precise temporal spans, and intelligible speech content, to a single, high-quality, timing-controlled audio output.

C Evaluation

C.1 Objective Metrics

We conduct a comprehensive objective evaluation to assess our model’s performance in two key areas: audio quality and semantic alignment with the text prompt.

Audio Quality. Our primary metric for audio fidelity is the Fréchet Audio Distance (FAD), which measures the distributional difference between generated and reference audio based on VGGish embeddings. To evaluate the consistency of acoustic event distributions, we also report Kullback-Leibler (KL) divergence computed using the PANNs tagging model. For completeness and comparison with prior works, we include the Inception Score (IS) and Fréchet Distance (FD) as supplementary metrics (Liu et al., 2023).

Semantic Alignment. To measure the alignment between the generated audio and its corresponding text prompt, we use the LAION-CLAP (Wu et al., 2023) score. This score is defined as the cosine similarity between the CLAP embeddings of the generated audio a and the text prompt t :

$$\text{CLAP}(a, t) = \frac{a \cdot t}{\|a\| \|t\|}. \quad (6)$$

A higher CLAP score indicates better semantic correspondence in the shared embedding space. All objective metrics are computed using the official AudioLDM evaluation toolkit for consistency.

C.2 Subjective Evaluation

For our subjective evaluation, we recruited 20 human evaluators to rate generated audio samples on a 5-point Mean Opinion Score (MOS) scale (1-5, with higher scores being better). The evaluation was divided into two distinct tasks, each with specific criteria:

Timing-Controlled Audio Generation. In this task, participants were presented with an audio clip and its corresponding timed prompt. They were asked to rate the audio based on the following two aspects:

- **Temporal Alignment (Temporal):** This measures the accuracy of timestamp adherence. The question asked was: "How accurately does the timing of the audio events match the given start and end times in the prompt?"
- **Overall Quality (OVL):** This assesses the perceptual quality of the audio clip itself. The question asked was: "Ignoring the prompt, how would you rate the overall quality and realism of the audio clip?"

Intelligible Audio Generation. In this task, participants were presented with an audio clip containing speech and the text it was intended to convey. They rated the audio based on the following three aspects:

- **Speech Intelligibility (Intelligible):** This measures the clarity of the spoken content. The question asked was: "How clear and understandable is the spoken content in the audio?"
- **Overall Quality (OVL):** This assesses the quality of the entire acoustic scene. The question asked was: "How would you rate the overall audio quality, including both the speech and any background sounds?"
- **Relevance (REL):** This measures the semantic correspondence between the audio and the text. The question asked was: "How well does the generated audio, as a whole, match the text description?"

D Training and Inference Pseudocode

Algorithms 1 and 2 summarize the training and inference procedures of ControlAudio. The training process follows our progressive three-stage design,

where the model gradually expands its controllability from text, to timing, and finally to phoneme-level speech content. During training, different condition combinations are randomly switched to preserve previously learned capabilities while introducing finer-grained control signals. During inference, we adopt a two-phase progressive sampling strategy, where temporal structure is first established with low guidance and the phoneme condition is then introduced with higher guidance to refine intelligible speech.

Algorithm 1: Progressive Training

Input: Training datasets $\mathcal{D}_1, \mathcal{D}_2, \mathcal{D}_3$
Output: Trained ControlAudio model ϵ_θ
Initialize diffusion model ϵ_θ and a pretrained text encoder τ_θ ;

Stage 1: Text-to-Audio Pretraining;
for each training step do
 Sample $(x, c_{\text{text}}) \sim \mathcal{D}_1$;
 Sample $t \sim \mathcal{U}(0, 1), \epsilon \sim \mathcal{N}(0, 1)$;
 $z_t \leftarrow \text{ForwardDiffusion}(x, t, \epsilon)$;
 $\hat{\epsilon} \leftarrow \epsilon_\theta(z_t, t, \tau_\theta(c_{\text{text}}))$;
 Update θ with $\|\epsilon - \hat{\epsilon}\|^2$;

Stage 2: Timing-Controlled Training;
for each training step do
 Sample $(x, c_{\text{text}}, c_{\text{timing}}) \sim \mathcal{D}_2$;
 Randomly choose condition
 $c \in \{c_{\text{text}}, (c_{\text{text}}, c_{\text{timing}})\}$;
 Sample t, ϵ ;
 $z_t \leftarrow \text{ForwardDiffusion}(x, t, \epsilon)$;
 $\hat{\epsilon} \leftarrow \epsilon_\theta(z_t, t, \tau_\theta(c))$;
 Update θ ;

Stage 3: Joint Training with Phoneme;
Unfreeze text encoder τ_θ ;
for each training step do
 Sample $(x, c_{\text{text}}, c_{\text{timing}}, c_{\text{phoneme}}) \sim \mathcal{D}_3$;
 Randomly choose condition
 $c \in \{c_{\text{text}}, (c_{\text{text}}, c_{\text{timing}}), (c_{\text{text}}, c_{\text{timing}}, c_{\text{phoneme}})\}$;
 Sample t, ϵ ;
 $z_t \leftarrow \text{ForwardDiffusion}(x, t, \epsilon)$;
 $\hat{\epsilon} \leftarrow \epsilon_\theta(z_t, t, \tau_\theta(c))$;
 Update θ ;

Algorithm 2: Progressive Sampling

Input: Text prompt c , timestep threshold t_1 , guidance scales $w_{\text{low}}, w_{\text{high}}$
Output: Generated audio sample x
Initialize $z_T \sim \mathcal{N}(0, I)$;
Construct: $c_1 = (c_{\text{text}}, c_{\text{timing}})$;
 $c_2 = (c_{\text{text}}, c_{\text{timing}}, c_{\text{phoneme}})$;
for $t = T$ to t_1 do
 $\hat{\epsilon} \leftarrow \epsilon_\theta(z_t, t, c_1)$;
 Apply classifier-free guidance with scale w_{low} ;
 $z_{t-1} \leftarrow \text{Denoise}(z_t, \hat{\epsilon})$;

for $t = t_1$ to 0 do
 $\hat{\epsilon} \leftarrow \epsilon_\theta(z_t, t, c_2)$;
 Apply classifier-free guidance with scale w_{high} ;
 $z_{t-1} \leftarrow \text{Denoise}(z_t, \hat{\epsilon})$;

Return generated audio $x = \text{Decode}(z_0)$;

Table 8: Category-wise performance on the AudioCondition test set.

Eb\uparrow											
Method	Alarm	Blender	Cat	Dishes	Dog	T-brush	Frying	Water	Speech	Cleaner	Avg
FreeAudio	40	47	14	20	23	78	60	51	39	72	44.34
ControlAudio	53.08	48.57	27.06	40.55	50.41	80.51	61.15	52.01	64.23	78.26	55.58
At\uparrow											
Method	Alarm	Blender	Cat	Dishes	Dog	T-brush	Frying	Water	Speech	Cleaner	Avg
FreeAudio	73	67	38	42	59	88	85	81	76	76	68.50
ControlAudio	87.22	69.39	41.25	67.48	85.50	91.67	91.38	82.69	88.65	90.00	79.52

Table 9: Sensitivity analysis of the transition timestep t_1 (coarse range).

t_1	100	95	90	85	80
FAD \downarrow	5.82	4.74	3.85	3.40	3.35
WER \downarrow	5.98	6.61	6.83	7.12	7.67

Table 10: Sensitivity analysis of the transition timestep t_1 (fine-grained range).

t_1	90	89	88	87	86	85
FAD \downarrow	3.85	3.61	3.52	3.53	3.47	3.40
WER \downarrow	6.83	6.85	6.84	6.86	7.01	7.12

E Detailed Results

E.1 Category-wise Results on AudioCondition

For the AudioCondition dataset, we provide detailed category-wise results in Table 8, including both event-based (Eb) and clip-level (At) metrics. ControlAudio consistently improves performance across most categories, particularly on speech-related and temporally complex events.

E.2 Sensitivity Analysis of Transition Timestep t_1

A clear trade-off between audio quality and speech intelligibility is observed: smaller t_1 improves FAD, while larger t_1 benefits WER. A fine-grained analysis around the selected setting further shows that the performance remains stable within a moderate range. Based on this trade-off, we set $t_1 = 88$ in all experiments.

F LLM Planning

F.1 Chain-of-Thought for Prompt Planning

Recent advances have demonstrated the powerful planning and cross-modal reasoning capabilities of large language models (LLMs) (Wang et al., 2025c, 2024). We leverage these capabilities by employing

an LLM to function as a "planner" that automatically converts a free-form natural language caption (y_c) into a precise, structured prompt (y_s) for our generative model. This conversion follows a three-stage reasoning process inspired by the Chain-of-Thought (CoT) paradigm, as illustrated in Figure 3. The process consists of the following steps:

- Event and Timing Planning.** Given the input caption, the LLM first identifies a set of distinct audio events $\mathcal{E} = \{e_i\}_{i=1}^N$. For each event, it infers a corresponding set of timing spans $\mathcal{T}_i = \{(s_{i1}, t_{i1}), \dots\}$, where s_{ik} and t_{ik} are the start and end times in seconds. This multi-span representation is designed to handle events that occur multiple times.
- Speech Content Planning.** For any event identified as speech ($e_i \in \mathcal{E}_{\text{speech}}$), the LLM then infers a plausible utterance c_i that fits the overall context. This step enriches the planned events with specific, intelligible speech content, resulting in a set of intermediate tuples $(e_i, \mathcal{T}_i, c_i)$.
- Prompt Recaption.** Finally, the LLM serializes the extracted information into the final structured prompt (y_s). This process starts with the original caption (y_c) and appends a specially formatted string for each planned event, which includes its name, associated time spans, and any inferred speech content.

F.2 Planning Results for AC-Filtered

To qualitatively assess the effectiveness of our LLM-based prompt planner, Table 11 presents several example results on samples from the AC-Filtered dataset. The table illustrates the planner's capability to parse complex, free-form captions (with or without associated speech text) and convert them into the precise, machine-readable structured prompts that our framework requires. This planning process is particularly crucial for enabling

complex, multi-speaker scenarios. For instance, the planner can generate prompts that assign different utterances to distinct speakers at specified times. This capability stands in sharp contrast to speech-oriented models like VoiceLDM, which, even when given a descriptive prompt about a conversation, can only render the entire speech content as a single utterance from one voice. This ability to plan and generate true dialogues is a key advantage of our approach for creating realistic acoustic scenes.

G Speech Transcription via ALM

To generate textual transcriptions for our segmented speech events, we utilize the Gemini 2.5 Pro model. Each clean audio segment is provided as direct input. We designed a prompt that serves a dual function: it instructs the model to accurately transcribe the spoken content while simultaneously acting as a quality filter. Specifically, the prompt directs the model to return an empty string if the speech in an audio segment is unintelligible or heavily obscured by noise, thereby automatically discarding low-quality samples. This process ensures that only clear, valid audio segments are converted into high-quality audio-text pairs. The full prompt used for this task is illustrated in Figure 5.

Table 11: Examples of LLM-based planning results, converting natural language inputs (caption and speech text) into structured prompts.

Input	Generated Structured Prompt (Output)
<p>Caption: She is talking in the park. Text: "Good morning! How are you feeling today?"</p>	<p>She is talking in the park. @ {park ambient sounds. & <0.00, 10.00>} @ {Female speech, woman speaking. & <1.50, 6.00> "Good morning! How are you feeling today?" }</p>
<p>Caption: A child yelling as a young boy talks during several slaps on a hard surface Text: "Say yeah, baby. Say yeah, baby. Are you over tired?"</p>	<p>A child yelling as a young boy talks during several slaps on a hard surface. @ {Young boy speaking & <1.50,8.00> "Say yeah, baby. Say yeah, baby. Are you over tired?" } @ {Child yelling & <2.00,6.00>} @ {slaps on a hard surface & <2.50,3.00> <5.00,5.50> }</p>
<p>Caption: A female speaking with some rustling followed by another female speaking Text: "The IT services at the King’s University College are proud to announce that we have launched"</p>	<p>A female speaking with some rustling followed by another female speaking. @ {Female speech, woman speaking & <0.50,6.00> "The IT services at the King’s University College are proud to announce that" } @ {rustling & <1.00,5.00>} @ {Female speech, woman speaking & <6.50,8.00> "we have launched" }</p>
<p>Caption: A duck quacks followed by a man talking while birds chirp in the distance Text: "Mama Mama snow mama come over here, baby"</p>	<p>A duck quacks followed by a man talking while birds chirp in the distance. @ {duck quack & <0.50,1.50>} @ {Man speaking & <2.00,7.50> "Mama Mama snow mama come over here, baby" } @ {birds chirping in the distance & <2.50,4.00> <5.50,7.00> }</p>
<p>Caption: Two men speaking with loud insects buzzing Text: "I’ve got gloves covered in mid repellent. Still fishing."</p>	<p>Two men speaking with loud insects buzzing. @ {Man speaking & <1.00,4.50> "I’ve got gloves covered in mid repellent." } @ {Man speaking & <5.00,6.50> "Still fishing." } @ {loud insects buzzing & <0.00,10.00> }</p>
<p>Caption: A man speaking as a stream of water splashes and flows while music faintly plays in the distance Text: "in the amateur show tonight then tomorrow on Saturday the broadcasters and the other amateur cast will be going out hope to do well there get some good footage hope you enjoy"</p>	<p>A man speaking as a stream of water splashes and flows while music faintly plays in the distance. @ {Man speaking & <0.50,9.50> "in the amateur show tonight then tomorrow on Saturday the broadcasters and the other amateur cast will be going out hope to do well there get some good footage hope you enjoy" } @ {water splashing and flowing & <0.00,10.00>} @ {faint music in the distance & <0.00,10.00> }</p>
<p>Caption: People are giggling, and a man speaks Text: (None)</p>	<p>People are giggling, and a man speaks. @ {people giggling & <1.00,5.00>} @ {Man speaking & <2.50,4.50> "What’s so funny?" }</p>
<p>Caption: (None) Text: "Some people talk about fucking the heads, but the way I do it, I just put my finger down there and pull it out."</p>	<p>A person is giving instructions or explaining a procedure. @ {Man speaking & <1.00,9.00> "Some people talk about fucking the heads, but the way I do it, I just put my finger down there and pull it out." }</p>

Gemini 2.5 pro for Speech Annotation

You are a strict speech transcription assistant.

Your task is to accurately transcribe the spoken content of the short audio segment provided. If the audio segment is mostly non-speech (e.g., music, noise) or the speech is unintelligible, you must return an empty transcription and provide a reason, as detailed in the guidelines. Always keep the language as spoken; do not translate.

Output must be a single, valid JSON object with no extra text.

Guidelines

Punctuation: Keep natural punctuation; do not paraphrase or add words.

Case & Numbers: Write as naturally spoken (e.g., proper nouns capitalized; numbers as spoken).

Multiple Speakers: If multiple speakers are present in the segment, keep it as a single transcript. You may minimally tag changes (e.g., [F]: ... [M]: ...) if necessary.

Non-speech or Unintelligible:

* If the segment is non-speech or too noisy: set "transcript": "" and "note": "non_speech" or "too_noisy".

* If only fragments are clear: transcribe only those parts. Never invent or guess words (hallucinate).

Language & Confidence: Provide a "lang" key (e.g., "en", "zh") and a subjective "confidence" key with a value between 0 and 1.

Strict JSON: Output only the JSON object. Do not add any introductory text or explanations.

Example of a successful transcription:

```
{
  "transcript": "Hello, how are you today?",
  "note": "",
  "lang": "en",
  "confidence": 0.98
}
```

Example of an unintelligible clip:

```
{
  "transcript": "",
  "note": "too_noisy",
  "lang": "",
  "confidence": 0.80
}
```

Figure 5: Gemini 2.5 pro for Speech Annotation.