SegTune: Structured and Fine-Grained Control for Song Generation

Anonymous ACL submission

Abstract

Recent advances in neural song generation have enabled high-quality synthesis from lyrics and global textual prompts. However, most systems fail to model temporally varying attributes of songs, severely limiting fine-grained control over musical structure and dynamics. To address this, we propose **SegTune**, a Diffusion Transformer-based framework enabling structured and fine-grained controllability by allowing users or large language models (LLMs) to specify local musical descriptions aligned to song segments. These segment prompts are temporally broadcast to corresponding time windows, while global prompts ensure stylistic coherence. To support precise lyric-to-music alignment, we introduce an LLM-based duration predictor that autoregressively generates sentence-level timestamps in LyRiCs format. We further construct a large-scale data pipeline for high-quality song collection with aligned lyrics and prompts, and propose new metrics to evaluate segment alignment and vocal consistency. Experiments demonstrate that SegTune outperforms existing baselines in both musicality and controllability. Visit our demo page for more generated songs of SegTune.

1 Introduction

Music and song constitute a powerful medium for emotional expression, combining linguistic content with rich acoustic and musical structures. Among music-related generative tasks, song generation is particularly challenging, as it requires the joint synthesis of vocals and accompaniment conditioned on lyrics and high-level control signals. Although commercial systems like Suno[1] have demonstrated expert-level performance, the open-source community for song generation still has considerable room for technical advancement.

Early song generation systems predominantly adopt autoregressive (AR) transformers to model long-range dependencies over quantized audio tokens. Representative approaches such as SongCreator (Lei et al., 2024) and MusiCoT (Lam et al., 2025) employ a shared token vocabulary for vocals and accompaniment, but this design introduces modality interference and limits expressive capacity. Subsequent works—including YuE (Yuan et al., 2025), SongGen (Liu et al., 2025b), and LeVo (Lei et al., 2025)—mitigate this issue by modeling vocals and accompaniment as separate token sequences. Nevertheless, AR methods remain computationally expensive and inflexible for interactive editing. Alternatively, non-autoregressive (NAR) frameworks—including DiffRhythm (Ziqian et al., 2025), DiffRhythm+ (Chen et al., 2025), ACE-Step (Gong et al., 2025) and JAM (Liu et al., 2025a)—adopt diffusion or flow-matching for accelerated generation. By operating in latent audio spaces, these methods significantly reduce inference time while maintaining reasonable audio fidelity. However, NAR models face inherent challenges: they compress the full song generation pipeline (composition and rendering) into a single latent diffusion process. As a result, they often struggle to jointly optimize musical structure, temporal coherence and voice-instrument balance.

A fundamental limitation shared by both AR and NAR song generation systems is their predominant reliance on global-only control signals. This limitation manifests in three interrelated ways. First, global prompts fail to capture inherent temporal dynamics—attributes of music, such as instrumentation, emotion, and energy naturally evolve across song segments, leading to homogeneous and mediocre outputs. Although some approaches incorporate coarse structural tags into lyrics (Yuan et al., 2025; Gong et al., 2025; Lei et al., 2025), they still lack the resolution for truly fine-grained and segment-level controls. Second, under global-only conditioning, jointly generating vocals and accompaniment imposes a substantial coordination bur-

[1] https://suno.com/home

den on the model, frequently leading to misaligned expression across modalities (Gong et al., 2025). Third, the absence of fine-grained control curtails expressive flexibility for both professional composers and amateur creators, which hampers practical usability for diverse creative workflows. These challenges are further exacerbated in some NAR models by their reliance on low-quality lyric durations—either zero-shot LLM-generated or manually specified by humans (Ziqian et al., 2025; Liu et al., 2025a). However, such duration annotations are not only time-consuming and error-prone, but also discourage user interaction.

To address these limitations, we propose SegTune, a NAR song generation framework that supports hierarchical control: a global prompt defines the overall style, while segment-level prompts—specifiable by users or auto-generated via an LLM—govern fine-grained per-segment attributes (e.g., emotion, rhythm, instrumentation). Specifically, we introduce a dedicated segment-level conditioning paradigm: a segment encoder injects fine-grained control signals into the corresponding temporal window of the latent sequence, while a global encoder preserves stylistic coherence across the entire song. Furthermore, we eliminate the need for manual lyric duration annotations by introducing a context-aware, LLM-based duration predictor, which adaptively generates sentence-level timestamps conditioned on the hierarchical prompts. Finally, these hierarchical textual conditionings and time-aligned lyrics embedding will guide the Diffusion Transformer (DiT) blocks to generate expressive and cohesively structured latent embeddings for both vocals and accompaniment. In summary, our contributions are as follows:

- We introduces a hierarchical, segment-level textual conditioning paradigm for fine-grained control in song generation.
- We develop an LLM-based duration predictor that generates sentence-level lyric timestamps, enabling accurate lyrics alignment without manual annotation.
- We construct a scalable pipeline to clean, annotate, and align high-quality songs with multi-level textual descriptions.
- We design new evaluation metrics, including segment-level MuLan score and singer attribute scoring, to rigorously assess fine-grained instruction following.

2 Related Work

2.1 Music Generation

Music generation focuses on producing coherent and stylistically consistent audio conditioned on text, melody, or other high-level cues. One prevalent approach is AR modeling, which sequentially generates discrete audio representations. MusicGen (Copet et al., 2023) combines residual vector quantization with transformer-based decoder to improve fidelity and controllability, while MusicLM (Agostinelli et al., 2023) adopts a two-stage architecture that first predicts semantic tokens before rendering acoustics. MeLoDy (Lam et al., 2023) enhances this design by replacing the acoustic decoder with a diffusion model to further improve synthesis quality and sampling efficiency.

In parallel, NAR methods (Chen et al., 2024; Evans et al., 2024a,b) have emerged as promising alternatives. These models typically employ diffusion (Ho et al., 2020; Rombach et al., 2022) or flow-matching (Lipman et al., 2023; Tong et al., 2024) mechanisms, operating in latent audio spaces to accelerate inference while maintaining high fidelity. Music ControlNet (Wu et al., 2024) extends this paradigm by introducing time-varying control signals—such as melody, rhythm, and dynamics—via temporally aligned conditioning, enabling fine-grained control over different aspects of the musical output. TVC-MusicGen (Yang et al., 2025b) adopts a segment-level diffusion framework for fine-grained music generation, sharing similarities with our paradigm, but its application is limited to instrumental music without vocal track.

2.2 Song Generation

Song generation aims to synthesize full musical compositions, including vocals and accompaniment, based on input lyrics and optional control signals. This task introduces additional challenges beyond instrumental music generation, such as aligning melody with lyrical phrasing, keeping vocal–accompaniment coherence, and maintaining musically meaningful transitions across segments.

Early systems such as Jukebox (Dhariwal et al., 2020) adopt AR transformers to model long-range dependencies over quantized audio tokens. Later works, like SongCreator (Lei et al., 2024) and MusiCoT (Lam et al., 2025), follow similar strategies but rely on a shared token vocabulary for both vocal and instrumental content, which introduces modality interference and limits expressive capac-

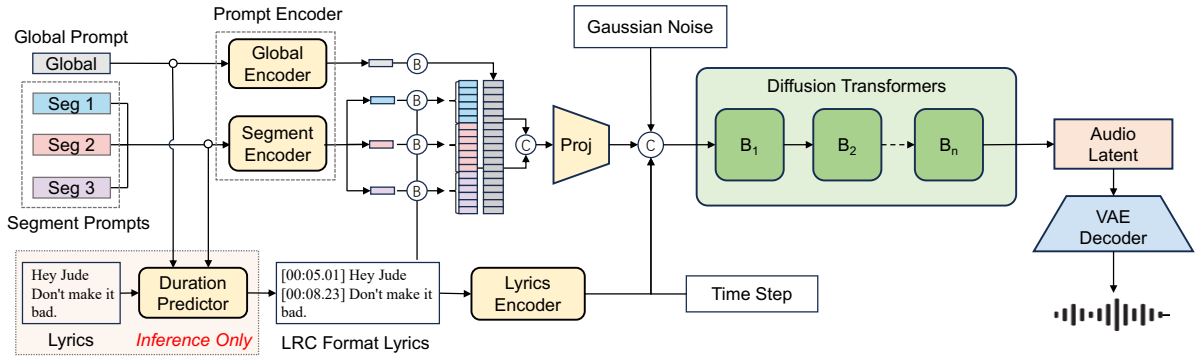

Figure 1: Overview of the SegTune architecture. The model takes lyrics and textual prompts as input. An LLM-based duration predictor estimates sentence-level durations for the lyrics, while a lyrics encoder embeds the lyrics and performs sentence-level alignment. The prompt encoder encodes both global and segment prompts into 1D vectors. The global prompt is broadcast to all time steps, whereas each segment prompt is broadcast to the frames within its corresponding temporal window, as determined by the duration predictor. All the conditional embeddings are concatenated and fed into a Diffusion Transformer. In the diagram, Ⓑ denotes temporal broadcasting within a segment window, and Ⓒ denotes channel-wise feature concatenation.

ity. To mitigate this, more recent systems, including SongGen (Liu et al., 2025b), YuE (Yuan et al., 2025) and LeVo (Lei et al., 2025), model vocals and accompaniment as separate token sequences, enabling more faithful synthesis and better exploitation of language models for long-context generation. However, AR approaches suffer from slow inference, high training complexity, and limited flexibility in downstream tasks like song editing.

Alternatively, NAR frameworks, including DiffRhythm, DiffRhythm+, ACE-Step, and JAM (Ziqian et al., 2025; Chen et al., 2025; Gong et al., 2025; Liu et al., 2025a), leverage diffusion-based architectures for faster generation and easier extension. Nevertheless, these systems still face challenges to maintain musicality, long-range coherence, and balancing vocals with accompaniment, since both composition and acoustic rendering are handled within only DiT.

In particular, existing systems rely primarily on global text prompts, which are insufficient to capture the temporal variability of real songs. Some works (Yuan et al., 2025; Gong et al., 2025; Lei et al., 2025) insert structural segmental labels in lyrics to align with predefined musical segments, yet finer-grained local control, such as adjusting the instrumentation or emotional intensity of specific segments, remains unsupported. In contrast, our work introduces fine-grained, segment-level textual conditioning along with an LLM-based duration prediction module. These designs enable precise control over the temporal evolution of musical attributes while supporting textual prompts for each segment, paving the way for more expressive and controllable song generation.

3 Methodology

3.1 Model Architecture

As illustrated in Figure 1, SegTune adopts a DiT (Peebles and Xie, 2023) architecture based on conditional flow-matching, extending previous song generation models (Gong et al., 2025; Ziqian et al., 2025; Chen et al., 2025; Liu et al., 2025a). The backbone consists of LLaMA-style (Touvron et al., 2023) transformer blocks. During training, a 1D VAE (Evans et al., 2024a) is used to compress raw audio with a sampling rate of 44 kHz into a latent sequence at 21.5 Hz, which serves as the target trajectory for flow supervision. SegTune conditions generation on three complementary sources: global textual prompts, segment-level textual prompts, and time-aligned lyrics. Together, these signals control both the semantic content and the temporal evolution of musical attributes.

In the following subsections, we first detail the hierarchical conditioning mechanism, followed by the duration predictor, which generates precise lyric timestamps, overcoming the reliance on manual annotations in prior DiT-based approaches.

3.1.1 Hierarchical prompts

SegTune supports text prompts at both the global and segment levels. The global prompt controls high-level song attributes such as genre, gender, timbre, and the overall emotional tone. In contrast, segment-level prompts explicitly describe time-varying attributes, including song structure

(e.g., verse or chorus), emotional transitions, rhythmic patterns, and instrumentation. This separation allows the model to disentangle global stylistic consistency from local musical variation. Unlike prior approaches that encode structural information implicitly within lyrics or coarse tags, SegTune introduces explicit segment-level textual prompts that are independently encoded and temporally injected into the diffusion process. Examples of global and segment prompts can be found in our demo page.

Specifically, each segment prompt is encoded by a segment encoder into a vector $\mathbf{e}_s^i \in \mathbb{R}^{1\times d_s}$, where i is the i-th segment. This vector is temporally broadcast to all latent frames within the corresponding segment time window, ensuring consistent local conditioning. Noted that each segment time window was provided by the duration predictor module during the inference stage (refers to section 3.1.2). Meanwhile, the global prompt is encoded by a global encoder and broadcast across all frames of the whole song. The global and segment embeddings are concatenated along the channel dimension and projected through a three-layer MLP to obtain the final conditioning embedding $E_{\text{text}} \in \mathbb{R}^{T\times d_{\text{text}}}$, where T is the length of the latent sequence and $d_{\text{text}} = 1024$. The detailed algorithm for text conditioning is shown in Appendix A.

We employ Qwen3-Embedding-0.6B (Zhang et al., 2025) as both the global and segment prompt encoder, as it preserves fine-grained semantic attributes in long-form textual descriptions. Details of hierarchical prompt construction are provided in Section 3.2. During inference, segment-level prompts can be specified by users or automatically generated by a large language model, enabling flexible and expressive controls.

3.1.2 Duration predictor

Accurate duration predictor is a critical component of SegTune. It determines segment boundaries for prompt broadcasting, aligns lyrics with audio latents in temporal dimension, and defines the initial noise length during inference.

Despite its importance, duration prediction has been largely overlooked. Prior NAR works either require error-prone manual timestamps (Gong et al., 2025; Ziqian et al., 2025; Chen et al., 2025) or use fragile zero-shot LLM prompting for word-level timing (e.g., JAM (Liu et al., 2025a)), neglecting the need for music-aware, lyric-aligned duration prediction module.

In this work, we fine-tune the Qwen3-4B-Base model (Yang et al., 2025a)—selected for its favorable capacity–speed trade-off—as a duration predictor that generates sentence-level timestamps in LyRiCs (LRC) format. Specifically, in the inference stage, given lyrics together with global and segment prompts, the predictor outputs timestamped lyrics, learning to align durations with musical attributes such as rhythm, emotion, genre, and lyric length. The instruction template is detailed in Appendix B, and the construction of ground-truth timestamps is described in Section 3.2. At inference time, segment temporal boundaries are derived directly from the predicted timestamps: for lyric-containing segments (e.g., verse and chorus), temporal boundaries are defined by the corresponding sentence intervals; for instrumental segments (e.g., intro, bridge and outro), boundaries are inferred from adjacent lyric segments, leveraging the structural contiguity inherent in typical song forms.

3.1.3 Lyric conditioning

To achieve precise, supervision-light phoneme-level alignment between lyrics and audio latent representations, we adopt the strategy of (Ziqian et al., 2025; Chen et al., 2025) that only need sentence-start annotations. Specifically, timestamped lyrics are directly available from the training data during training. In inference stage, the duration predictor automatically estimates these start times of lyrics. Each lyric sentence is first converted into a phoneme sequence via a grapheme-to-phoneme model. Then, a placeholder sequence E_{lyrics}, matching the length of the audio latent sequence E_{audio}, is initialized with <*pad*> tokens. The phoneme sequence is written into E_{lyrics} starting at the frame of the lyric's start time.

Finally, the textual prompt embedding E_{text}, along with E_{lyrics}, E_{audio}, and the time-step embedding E_t (broadcast to length T) are concatenated along the channel dimension and fed into the DiT.

3.2 Data Pipeline

SegTune is trained on a curated internal corpus of mainly Chinese pop songs and a small amount of other language songs. To ensure high-fidelity segment annotations and precise lyric-level timestamping, we devise a dedicated three-stage data curation pipeline, and the workflow is shown in the Appendix C.

3.2.1 Quality filtering

We first apply metadata-based filtering (duration, sampling rate, channels, energy, etc.) and sound

event detection to discard non-musical clips. Subsequently, we leverage Audiobox (Tjandra et al., 2025) and SongEval (Yao et al., 2025) to score audio aesthetics and prune low-quality samples.

3.2.2 Lyrics processing

For songs without lyrics, we separate vocals using Demucs v4 (Rouard et al., 2023), then transcribe them with FireRedASR (Xu et al., 2025b) (Mandarin) or Whisper-Large-v3 (Radford et al., 2022) (other languages). When ground-truth LRC files are available, we first remove non-lyrics metadata using an LLM-based filter, then validate the cleaned lyrics against ASR outputs via edit distance—discarding samples with high discrepancy. Then, structural labels (e.g., *intro*, *verse*, *chorus*) are extracted using the all-in-one music understanding model (Kim and Nam, 2023).

3.2.3 Hierarchical prompt annotation

Global and segment prompts are generated via Audio Flamingo 3 (Goel et al., 2025), with the structural label prepended to each segment prompt to enable controllability. System prompt templates used for Audio Flamingo 3 are detailed in the Appendix D. To mark boundaries, fixed prompts—"This piece is the start/end of the song."—are assigned to the first and last 0.5 s of each sample.

3.3 Training and Inference

The model is trained under the Conditional Flow Matching (CFM) framework, which aims to learn a function $v_\theta(t, C, x_t)$ that approximates the flow $u(x_t|x_0, x_1)$. The training objective is defined as:

$$\mathcal{L} = \mathbb{E}_{t,q,p} \left\| v_\theta(t, C, x_t) - u(x_t \mid x_0, x_1) \right\|^2, \quad (1)$$

$$x_t = (1-t)x_0 + tx_1, \quad (2)$$

$$u(x_t \mid x_0, x_1) = x_1 - x_0, \quad (3)$$

where $x_0 \sim p(x_0)$ represents a sample from the prior distribution $\mathcal{N}(0, \mathbf{I})$. $x_1 \sim q(x_1)$ is drawn from the target data distribution, $t \sim \mathcal{U}(0, 1)$ denotes the diffusion time step, and C denotes the conditioning input, which includes lyrics and textual prompts. The target vector $u(x_t \mid x_0, x_1)$ represents the flow at x_t.

Specifically, the training procedure follows a three-stage paradigm. (i) **Pre-training**: we retain songs with $\geq$32 kHz sampling rate, durations of 30s to 6 mins, and audio aesthetics scores above the 5th percentile, yielding approximately 370k songs (around 27k hours). (ii) **Fine-tuning**: stricter criteria apply—44 kHz, stereo, and top-50% audio aesthetics scores—resulting in about 50k songs (around 4k hours). (iii) **Preference Alignment**: following prior work (Lei et al., 2025; Chen et al., 2025; Liu et al., 2025a), we adopt iterative Direct Preference Optimization (DPO) (Rafailov et al., 2023) with two rounds. Starting from the SFT model, each round generates 16 candidates per lyric; win–loss pairs are formed by selecting pairs with SongEval (Yao et al., 2025) score differences above a threshold, where the winning sample exceeds the 75th percentile of all candidates. Each DPO round uses around 20k such pairs.

During inference, we employ the Euler ODE solver. For Classifier-Free Guidance (CFG), we use the formulation:

$$v = v_u + \text{cfg}(v_c - v_u) - \text{cfg}_n(v_n - v_u), \quad (4)$$

where v_u and v_c denote the unconditional and conditional flows, respectively, and v_n represents the flow obtained under negative conditions (Ban et al., 2024). In the negative condition setup, the lyric conditioning is removed, while both global and local prompts are replaced with negative prompts. Empirically, we set $\text{cfg} = 3$ and $\text{cfg}_n = 1$.

4 Experiments

4.1 Experimental Setup

SegTune is trained on an internal corpus of predominantly Mandarin pop songs (>90%), spanning diverse artists, lyrical themes, and song structures. The diffusion backbone is a DiT-style architecture (1.1B parameters, 16 LLaMA-style decoder blocks), following (Ziqian et al., 2025). Training proceeds in three stages: (i) 20-epoch pretraining with batch size = 32, lr = 2e-5, (ii) 8-epoch fine-tuning with the same setting, and (iii) two rounds of iterative DPO (4 epochs each, batch size = 8, grad accumulation = 4, and lr = 5e-7). During training, 20% dropout is applied independently to global and segment-level conditions to enable classifier-free guidance. We also augment global and segment text prompts using LLM-based rewriting to enhance generalization across diverse real-world text input styles.

For the duration predictor, we fine-tune Qwen3-4B-Base on >100k LRC-formatted lyrics for 8 epochs (batch size = 8, grad accumulation = 4, max new tokens = 4096, and lr = 2e-5), using LoRA (Hu et al., 2022) (rank = 32) for efficiency.

Table 1: Performance comparison of SegTune and baseline models on objective metrics. SegTune-SFT denotes the model that has undergone both pretraining and supervised fine-tuning (SFT), while SegTune-DPO refers to the model further refined via 2 iterations of Direct Preference Optimization (DPO) starting from the SegTune-SFT checkpoint. G-Mulan denotes Global Mulan score, and S-Mulan denotes Segment Mulan score.

Models	PER↓	AudioBox-aesthetic↑				SongEval↑					Instruction-following↑		
		CE	CU	PC	PQ	Coh	Mem	NVBP	CSS	OM	G-Mulan	Gender	Age
YuE	48.5%	7.16	7.66	6.27	8.09	3.51	3.27	3.22	3.26	3.22	0.29	80.7%	44%
LeVo	29.8%	7.43	7.71	5.25	8.29	3.46	3.29	3.20	3.29	3.35	0.32	90.6%	50%
DiffR.+	27.4%	7.55	7.80	6.72	8.21	4.05	3.84	3.65	3.82	3.76	**0.47**	37.5%	54%
ACE-Step	35.6%	7.38	7.53	6.71	7.88	3.98	3.78	3.65	3.77	3.74	0.35	78.1%	56%
SegTune-SFT	**14.5%**	7.38	7.71	**6.83**	8.23	3.54	3.22	3.23	3.32	3.19	**0.47**	**96.7%**	**57%**
SegTune-DPO	18.5%	**7.63**	**7.85**	6.80	**8.36**	**4.25**	**4.06**	**4.09**	**4.08**	**3.97**	0.46	81.0%	51%

4.2 Baselines and Evaluation

We select four representative state-of-the-art baselines: YuE (Yuan et al., 2025) and LeVo (Lei et al., 2025), which adopt AR language models; and DiffRhythm+ (Chen et al., 2025) and ACE-Step (Gong et al., 2025), which are diffusion-based models. All baselines support song generation conditioned on lyrics and global textual prompts. And the test set for song generation consists of 15 Mandarin pop songs generated by ChatGPT, including their lyrics and associated prompts.

We evaluate aesthetic quality using objective metrics: (i) Phoneme Error Rate (PER), evaluating the intelligibility and lyrical fidelity of songs. FireRedASR (Xu et al., 2025b) was used to transcribe generated songs; (ii) Audiobox-Aesthetics (Tjandra et al., 2025), assessing production quality (PQ), production complexity (PC), content enjoyment (CE), and content usefulness (CU); (iii) SongEval (Yao et al., 2025), measuring coherence (Coh), memorability (Mem), natural vocal breathing/phrasing (NVBP), clarity of song structure (CSS), and overall musicality (OM).

The instruction-following capability of music generation models is then evaluated using the Muq-MuLan (Zhu et al., 2025) score. Muq-Mulan is a widely adopted multimodal representation model that aligns text descriptions and music clips through contrastive learning, and has been extensively used for music evaluation (Gong et al., 2025; Yuan et al., 2025). Global Mulan score measures the global alignment between songs and their global prompts. In addition, we compute the Segment MuLan scores for individual song segments based on segment prompts and use their average to assess the model's segment instruction adherence. We further evaluate the model's instruction-following accuracy on singer-related attributes—gender and age—since MuLan shows limited sensitivity to vocal identity. Prompts are selected from the test set, and the singer's gender and age (teenager, 20s, and 40s) are modified to form new global prompts. For gender accuracy, we directly employ Qwen3-Omni-30B-A3B-Captioner (Xu et al., 2025a) to assess the gender of generated songs. For age accuracy, we randomly sample two generated songs and conduct an A/B test, where Qwen3-Omni determines which song corresponds to the older singer. The final accuracy is computed based on the correctness of these pairwise judgments.

For subjective evaluation, we conducted 5-scale Mean Opinion Score (MOS) listening tests, in which five listeners rated each sample on musicality and quality using fine-grained evaluation criteria. To ensure balanced evaluation, every listener assessed all 5 system outputs for each of 9 songs—totaling 45 ratings per listener.

5 Results and Discussions

5.1 Results of Objective Evaluation

Table 1 presents objective results for SegTune (SFT and DPO stages) and baselines. SegTune achieves strong performance overall: it has the lowest PER, indicating superior lyrical fidelity and vocal intelligibility. In contrast, we noticed that the vocal generated by YuE is a bit hoarse. Both YuE and ACE-Step frequently interpolate past lyrics into current segments, resulting in notably higher PER. As for AudioBox-Aesthetics and SongEval metrics, SegTune-SFT is competitive with baselines, and SegTune-DPO further enhances quality—particularly in SongEval metrics.

Since baseline models support only global control, we assess instruction-following via global MuLan scores and singer attribute control accuracy (gender and age). SegTune-SFT attains the highest

gender control accuracy while preserving strong global MuLan alignment, demonstrating superior instruction-following capability.

After DPO fine-tuning, we observe a trade-off: the musical quality improves, yet the instruction-following degrades, especially for age and gender. This stems from bias in the preference data and DPO constrains: winning songs scored by SongEval are dominated from young female vocals, and DPO optimizes only for preference alignment without enforcing age and gender fidelity. One promising direction is to adopt online policy optimization, wherein generated songs are dynamically evaluated for demographic fidelity, and deviations from user-specified attributes are explicitly penalized, which will be investigated in the future work.

5.2 Results of Subjective Evaluation

As shown in Figure 2, SegTune-DPO achieves the highest MOS for musicality among all baselines, with the lowest standard deviation (4.57 ± 0.52)—indicating not only superior musical expressiveness but also exceptional consistency across samples. This gain is attributable to the DPO fine-tuning stage, which effectively suppresses musically degraded outputs. In contrast, YuE and LeVo exhibit higher variability in Musicality scores (standard deviations > 0.9), as they heavily rely on audio prompts for conditioning, while the use of text-based global prompts leads to a marked degradation in accompaniment coherence and diversity.

As for quality, SegTune-DPO achieves the second-highest results, while exhibiting smallest standard deviation (3.87 ± 0.56), surpassed only by LeVo (3.96 ± 0.87). This is likely attributable to our data pipeline, which selects high quality training data, and to the post-training stages, which effectively enhances audio fidelity.

5.3 Ablation Studies for Prompt Encoder

Since SegTune differs from baselines in data, scale, and architecture, the gains in Table 1 may reflect confounding factors. To isolate the impact of segment prompt injection, we perform ablation studies with identical data, backbone, and hyperparameters—varying only the prompt encoder.

Specifically, we examine two design dimensions: (i) composition strategies for integrating embeddings from the hierarchical prompt encoders—namely, global-only, concatenate, or mixed settings; and (ii) the choice of prompt encoder backbone—comparing Qwen3-Embedding,

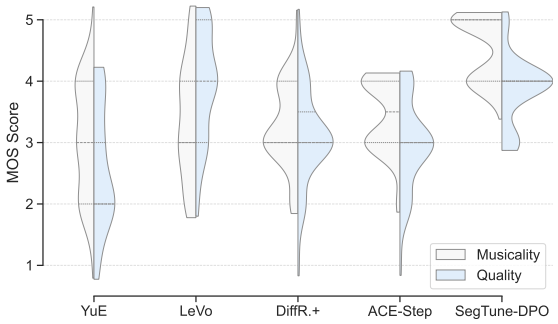

Figure 2: Violin plots of MOS results for musicality and quality evaluation.

a state-of-the-art textual encoder optimized for long-form, natural-language prompts, against Muq-MuLan (Zhu et al., 2025), the music-specialized multimodal encoder widely adopted in recent song generation systems (Chen et al., 2025; Liu et al., 2025a; Yang et al., 2025b).

5.3.1 Global-only setting

We train variants of SegTune-SFT using only global prompts with either Qwen3-Embedding or Muq-MuLan (Zhu et al., 2025) as the global encoder. As Table 2 shows, Qwen3-Embedding consistently surpasses Muq-MuLan in both musicality and instruction-following. Notably, gender control accuracy reaches 92.2% with Qwen3-Embedding, versus near-chance performance with Muq-MuLan. This discrepancy stems primarily from the fact that during MuQ-Mulan's text–music alignment training, singer-related attributes were not included in music captions, resulting in a poor representation of vocal characteristics in its embeddings. We conduct controlled experiments and visualization analyses on singer gender control to further elucidate the limitations of MuQ-MuLan in Appendix E.

5.3.2 Concatenate setting

In this setting, global and segment embeddings are concatenated along the channel dimension. Since Qwen3-Embedding achieves high-fidelity control over singer attributes, we fix it as global encoder and compare two choices for segment encoder: Muq-MuLan (Zhu et al., 2025) and Qwen3-Embedding. Although Muq-MuLan encodes singer-related characteristics weakly, it remains suitable for segment-level control—where prompts primarily govern musical attributes such as instrumentation, emotion, and rhythm. In Table 2: (i) the two variants achieve comparable overall musicality, yet the configuration using Qwen3-

Table 2: Impact of prompt encoder settings on objective performance of SegTune at the SFT stage. MuQ refers to MuQ-Mulan encoder, and Qwen3. refers to Qwen3-Embedding. G-Mulan denotes Global Mulan score, and S-Mulan denotes Segment Mulan score.

Prompt encoders		**AudioBox-aesthetic↑**				**SongEval↑**					**Instruction-following↑**			
Global	**Segment**	**CE**	**CU**	**PC**	**PQ**	**Coh**	**Mem**	**NVBP**	**CSS**	**OM**	**G-Mulan**	**S-Mulan**	**Gender**	**Age**
MuQ	–	7.42	7.60	6.63	8.19	3.18	2.87	2.81	2.93	2.86	0.39	0.30	47.6%	47%
Qwen3.	–	7.39	7.64	6.76	8.19	3.42	3.11	3.11	3.20	3.12	0.40	0.33	92.2%	50%
Concat.														
Qwen3.	MuQ	**7.57**	**7.82**	6.63	**8.35**	**3.62**	**3.37**	**3.30**	**3.43**	**3.34**	0.44	0.37	84.4%	46%
Qwen3.	Qwen3.	7.38	7.71	**6.83**	8.23	3.54	3.22	3.23	3.32	3.19	**0.47**	**0.38**	**96.7%**	**57%**
Mixed														
Qwen3.	Qwen3.	7.29	7.73	6.33	8.32	3.43	3.14	3.15	3.23	3.12	0.43	0.35	90.5%	**57%**

Table 3: Impact of duration predictor on SegTune-DPO performance. MAE denotes the mean absolute error (in seconds) of sentence-level duration prediction. GT means using the ground truth timestamps of lyrics for model inference. G-Mulan denotes Global Mulan score, and S-Mulan denotes Segment Mulan score.

Predictor	**MAE↓**	**AudioBox-aesthetic↑**				**SongEval↑**					**Instruction-following↑**			
		CE	**CU**	**PC**	**PQ**	**Coh**	**Mem**	**NVBP**	**CSS**	**OM**	**G-Mulan**	**S-Mulan**	**Gender**	**Age**
GT	0.00	7.65	7.74	6.58	8.35	**4.33**	4.16	4.17	4.12	4.01	0.45	0.47	81.9%	61%
Qwen3-SFT	0.99	7.66	7.74	**6.58**	**8.36**	4.32	**4.16**	**4.18**	**4.16**	**4.06**	0.45	0.41	81.9%	61%
GPT-4o	3.24	**7.69**	**7.76**	6.29	8.34	4.19	4.00	4.08	3.98	3.86	0.42	0.41	81.9%	58%

Embedding for the segment encoder yields stronger instruction-following performance; (ii) all objective musicality metrics significantly surpass those of the global-only setting. These results validate the superiority of hierarchical prompts in SegTune.

5.3.3 Mixed setting

Following (Jiang et al., 2025), we also explore linearly fuse global and segment embeddings (with weights 0.2 and 0.8), but observe clear drops in musicality and instruction following—likely due to semantic interference from mixing, unlike the cleaner separation in concatenation.

5.4 Ablation Studies for Duration Predictor

An accurate duration predictor yields more musically plausible lyric timestamps than user-provided ones, lowering user expertise barriers. We evaluate two approaches: (i) a Qwen3-4B-Base model fine-tuned on our dataset to predict per-line start timestamps; and (ii) a zero-shot GPT-4o predictor with engineered prompting, following JAM (Liu et al., 2025a). This evaluation uses 15 real-world Mandarin pop songs with paired lyrics and prompts, to enable direct comparison with ground-truth timestamps. To ensure fairness, these songs are excluded from the training set.

As shown in Table 3, the Qwen3-SFT predictor achieves a mean absolute error (MAE) of 0.99s, significantly lower than GPT-4o's. It also matches or exceeds GPT-4o across nearly all musicality dimensions, with comparable scores only in Content Enjoyment and Usefulness. Segment MuLan and gender control remain largely unaffected, indicating robustness to local timing variations.

6 Conclusions

We presented SegTune, a NAR framework for controllable song generation. By jointly leveraging segment-level textual conditioning and LLM-based duration predictor, it enables fine-grained control over emotion, rhythm, and instrumentation while preserving global coherence. Coupled with a scalable data pipeline and comprehensive evaluation, SegTune significantly advances controllability and musical quality over existing open-source systems. Although our experiments focus on Mandarin pop songs, SegTune is language-agnostic and can be extended to other languages given appropriate data.

In the future work, we will explore: (i) supporting richer local interactions—e.g., multi-singer transitions—currently limited by data scarcity; and (ii) integrating SegTune into a conversational agent, which can interpret user intent and dynamically generate lyrics and hierarchical prompts for human-machine collaborative creation.

7 Limitations

While SegTune advances fine-grained controllability in non-autoregressive song generation, several limitations remain.

First, SegTune is trained on ground-truth songs with clearly defined segment structures. Consequently, when the segment structure of the user input is ambiguous or poorly specified, the performance of the duration predictor degrades, which will in turn negatively affect the overall song generation quality.

Second, while the hierarchical prompting scheme allows control over segment-level attributes such as emotion and instrumentation, it cannot model fine-grained intra-segment dynamics (e.g., gradual crescendos or vocal ornamentation).

We view these limitations not as fundamental flaws of SegTune, but as clear pathways for future work—such as integrating end-to-end structure prediction and developing multi-objective alignment strategies that preserve both musicality and user-specified attributes.

References

Andrea Agostinelli, Timo I. Denk, Zalán Borsos, Jesse Engel, Mauro Verzetti, Antoine Caillon, Qingqing Huang, Aren Jansen, Adam Roberts, Marco Tagliasacchi, Matt Sharifi, Neil Zeghidour, and Christian Frank. 2023. Musiclm: Generating music from text. *Preprint*, arXiv:2301.11325.

Yuanhao Ban, Ruochen Wang, Tianyi Zhou, Minhao Cheng, Boqing Gong, and Cho-Jui Hsieh. 2024. Understanding the impact of negative prompts: When and how do they take effect? In *Computer Vision – ECCV 2024: 18th European Conference, Milan, Italy, September 29–October 4, 2024, Proceedings, Part LXXXIX*, page 190–206, Berlin, Heidelberg. Springer-Verlag.

Huakang Chen, Yuepeng Jiang, Guobin Ma, Chunbo Hao, Shuai Wang, Jixun Yao, Ziqian Ning, Meng Meng, Jian Luan, and Lei Xie. 2025. Diffrhythm+: Controllable and flexible full-length song generation with preference optimization. *Preprint*, arXiv:2507.12890.

Ke Chen, Yusong Wu, Haohe Liu, Marianna Nezhurina, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. 2024. Musicldm: Enhancing novelty in text-to-music generation using beat-synchronous mixup strategies. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1206–1210.

Jade Copet, Felix Kreuk, Itai Gat, Tal Remez, David Kant, Gabriel Synnaeve, Yossi Adi, and Alexandre Défossez. 2023. Simple and controllable music generation. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Prafulla Dhariwal, Heewoo Jun, Christine Payne, Jong Wook Kim, Alec Radford, and Ilya Sutskever. 2020. Jukebox: A generative model for music. *arXiv preprint arXiv:2005.00341*.

Zach Evans, Julian D. Parker, CJ Carr, Zachary Zukowski, Josiah Taylor, and Jordi Pons. 2024a. Long-form music generation with latent diffusion. In *Proceedings of the 25th International Society for Music Information Retrieval Conference, ISMIR 2024, San Francisco, California, USA and Online, November 10-14, 2024*, pages 429–437.

Zach Evans, Julian D. Parker, CJ Carr, Zack Zukowski, Josiah Taylor, and Jordi Pons. 2024b. Stable audio open. *Preprint*, arXiv:2407.14358.

Arushi Goel, Sreyan Ghosh, Jaehyeon Kim, Sonal Kumar, Zhifeng Kong, Sang gil Lee, Chao-Han Huck Yang, Ramani Duraiswami, Dinesh Manocha, Rafael Valle, and Bryan Catanzaro. 2025. Audio flamingo 3: Advancing audio intelligence with fully open large audio language models. *Preprint*, arXiv:2507.08128.

Junmin Gong, Sean Zhao, Sen Wang, Shengyuan Xu, and Joe Guo. 2025. Ace-step: A step towards music generation foundation model. *Preprint*, arXiv:2506.00045.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems*, volume 33, pages 6840–6851. Curran Associates, Inc.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

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*.

Taejun Kim and Juhan Nam. 2023. All-in-one metrical and functional structure analysis with neighborhood attentions on demixed audio. In *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*.

Max W. Y. Lam, Qiao Tian, Tang Li, Zongyu Yin, Siyuan Feng, Ming Tu, Yuliang Ji, Rui Xia, Mingbo Ma, Xuchen Song, Jitong Chen, Yuping Wang, and Yuxuan Wang. 2023. Efficient neural music generation. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Max W. Y. Lam, Yijin Xing, Weiya You, Jingcheng Wu, Zongyu Yin, Fuqiang Jiang, Hangyu Liu, Feng Liu, Xingda Li, Wei-Tsung Lu, Hanyu Chen, Tong

Feng, Tianwei Zhao, Chien-Hung Liu, Xuchen Song, Yang Li, and Yahui Zhou. 2025. Analyzable chain-of-musical-thought prompting for high-fidelity music generation. *Preprint*, arXiv:2503.19611.

Shun Lei, Yaoxun Xu, Zhiwei Lin, Huaicheng Zhang, Wei Tan, Hangting Chen, Jianwei Yu, Yixuan Zhang, Chenyu Yang, Haina Zhu, Shuai Wang, Zhiyong Wu, and Dong Yu. 2025. Levo: High-quality song generation with multi-preference alignment. *arXiv preprint arXiv:2506.07520*.

Shun Lei, Yixuan Zhou, Boshi Tang, Max W. Y. Lam, Feng liu, Hangyu Liu, Jingcheng Wu, Shiyin Kang, Zhiyong Wu, and Helen M. Meng. 2024. Songcreator: Lyrics-based universal song generation. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Yaron Lipman, Ricky T. Q. Chen, Heli Ben-Hamu, Maximilian Nickel, and Matthew Le. 2023. Flow matching for generative modeling. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Renhang Liu, Chia-Yu Hung, Navonil Majumder, Taylor Gautreaux, Amir Ali Bagherzadeh, Chuan Li, Dorien Herremans, and Soujanya Poria. 2025a. Jam: A tiny flow-based song generator with fine-grained controllability and aesthetic alignment. *Preprint*, arXiv:2507.20880.

Zihan Liu, Shuangrui Ding, Zhixiong Zhang, Xiaoyi Dong, Pan Zhang, Yuhang Zang, Yuhang Cao, Dahua Lin, and Jiaqi Wang. 2025b. Songgen: A single stage auto-regressive transformer for text-to-song generation. In *Forty-second International Conference on Machine Learning*.

William Peebles and Saining Xie. 2023. Scalable diffusion models with transformers. In *IEEE/CVF International Conference on Computer Vision, ICCV 2023, Paris, France, October 1-6, 2023*, pages 4172–4182. IEEE.

Alec Radford, Jong Wook Kim, Tao Xu, Greg Brockman, Christine McLeavey, and Ilya Sutskever. 2022. Robust speech recognition via large-scale weak supervision. *Preprint*, arXiv:2212.04356.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea Finn. 2023. Direct preference optimization: Your language model is secretly a reward model. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2022. High-resolution image synthesis with latent diffusion models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2022, New Orleans, LA, USA, June 18-24, 2022*, pages 10674–10685. IEEE.

Simon Rouard, Francisco Massa, and Alexandre Défossez. 2023. Hybrid transformers for music source separation. In *ICASSP 23*.

Andros Tjandra, Yi-Chiao Wu, Baishan Guo, John Hoffman, Brian Ellis, Apoorv Vyas, Bowen Shi, Sanyuan Chen, Matt Le, Nick Zacharov, and 1 others. 2025. Meta audiobox aesthetics: Unified automatic quality assessment for speech, music, and sound. *arXiv preprint arXiv:2502.05139*.

Alexander Tong, Kilian Fatras, Nikolay Malkin, Guillaume Huguet, Yanlei Zhang, Jarrid Rector-Brooks, Guy Wolf, and Yoshua Bengio. 2024. Improving and generalizing flow-based generative models with minibatch optimal transport. *Trans. Mach. Learn. Res.*, 2024.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023. Llama: Open and efficient foundation language models. *Preprint*, arXiv:2302.13971.

Shih-Lun Wu, Chris Donahue, Shinji Watanabe, and Nicholas J. Bryan. 2024. Music controlnet: Multiple time-varying controls for music generation. *IEEE/ACM Trans. Audio, Speech and Lang. Proc.*, 32:2692–2703.

Jin Xu, Zhifang Guo, Hangrui Hu, and Yunfei Chu et al. 2025a. Qwen3-omni technical report. *arXiv preprint arXiv:2509.17765*.

Kai-Tuo Xu, Feng-Long Xie, Xu Tang, and Yao Hu. 2025b. Fireredasr: Open-source industrial-grade mandarin speech recognition models from encoder-decoder to llm integration. *arXiv preprint arXiv:2501.14350*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, and Bowen Yu et al. 2025a. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Chenyu Yang, Hangting Chen, Shuai Wang, Haina Zhu, and Haizhou Li. 2025b. Tvc-musicgen: Time-varying structure control for background music generation via self-supervised training. In *Annual Conference of the International Speech Communication Association*, pages 1238–1242.

Jixun Yao, Guobin Ma, Huixin Xue, Huakang Chen, Chunbo Hao, Yuepeng Jiang, Haohe Liu, Ruibin Yuan, Jin Xu, Wei Xue, and 1 others. 2025. Songeval: A benchmark dataset for song aesthetics evaluation. *arXiv preprint arXiv:2505.10793*.

Ruibin Yuan, Hanfeng Lin, Shuyue Guo, Ge Zhang, Jiahao Pan, Yongyi Zang, and Haohe Liu et al. 2025. Yue: Scaling open foundation models for long-form music generation. *Preprint*, arXiv:2503.08638.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. 2025. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *Preprint*, arXiv:2506.05176.

Haina Zhu, Yizhi Zhou, Hangting Chen, Jianwei Yu, Ziyang Ma, Rongzhi Gu, Yi Luo, Wei Tan, and Xie Chen. 2025. Muq: Self-supervised music representation learning with mel residual vector quantization. *Preprint*, arXiv:2501.01108.

Ning Ziqian, Chen Huakang, Jiang Yuepeng, Hao Chunbo, Ma Guobin, Wang Shuai, Yao Jixun, and Xie Lei. 2025. DiffRhythm: Blazingly fast and embarrassingly simple end-to-end full-length song generation with latent diffusion. *arXiv preprint arXiv:2503.01183*.

A Algorithm for Global and Segment Text Conditioning

The algorithm 1 and algorithm 2 for extracting global and segment-level text conditioning differs slightly between training and inference stages. During training, the start time of each lyric line and the temporal boundaries of each segment are pre-annotated by the data pipeline and thus directly available for model training. In contrast, during inference, the input lyrics provided by the user to SegTune contain no timestamps. Consequently, the duration predictor module is employed to estimate the start times of individual lyric lines and the time spans of each segment.

B Prompt of Duration Predictor

Input Prompt to Qwen3-4B-Base

You are a professional music composer and vocal arranger. Your task:
1. Analyze the lyrics and the song description below.
2. For each line of lyrics, estimate a reasonable singing duration. Base your estimation jointly on:

- The intrinsic characteristics of the line itself (e.g., length, phrasing, complexity)
- The overall song attributes;
- The structural flow of the song, including instrumental breaks, natural pauses, and transitions;

3. Return: Output a complete '.lrc' style list

Algorithm 1 Global and Segment Text Conditioning Embedding Extraction during Training Stage

Require: Global textual prompt x_g, segment textual prompts and the corresponding time boundary $\{(x_s^i, t_s^i, t_e^i)\}_{i=1}^N$, where i represent the i-th segment, t_s and t_e denote the start and end time of the i-th segment.

Require: Sampling rate r, downsample rate r_d, number of latent frames T, global encoder f_g, segment encoder f_s, output projection `out_proj`

$\mathbf{e}_g \leftarrow f_g(x_g) \in \mathbb{R}^{1\times d_g}$

$E_g \leftarrow \texttt{repeat}(\mathbf{e}_g, T) \in \mathbb{R}^{T\times d_g}$ // Broadcast the global prompt across all time frames.

Initialize $E_s \in \mathbb{R}^{T\times d_s}$ with zeros

for each (x_s^i, t_s^i, t_e^i) in segment prompts **do**

$j_s^i \leftarrow \lfloor t_s^i \cdot r/r_d \rfloor$, $i_e^i \leftarrow \lfloor t_e^i \cdot r/r_d \rfloor$

$\mathbf{e}_s^i \leftarrow f_s(x_s^i) \in \mathbb{R}^{1\times d_s}$

$E_s[j_s^i : j_e^i] \leftarrow \mathbf{e}_s^i$ // Broadcast each segment prompt to its corresponding temporal window.

end for

$E_{\text{cat}} \leftarrow \texttt{concat}(E_g, E_s, \text{dim} = -1)$, $E_{\text{cat}} \in \mathbb{R}^{T\times(d_g+d_s)}$

$E_{text} \leftarrow \texttt{out\_proj}(E_{\text{cat}})$

return fused embedding $E_{text} \in \mathbb{R}^{T\times d_{text}}$

Algorithm 2 Global and Segment Text Conditioning Embedding Extraction during Inference Stage

Require: Global textual prompt x_g, segment textual prompts $\{x_s^i\}_{i=1}^N$, where i represent the i-th segment.

Require: Sampling rate r, downsample rate r_d, global encoder f_g, segment encoder f_s, duration predictor module f_p, output projection `out_proj`

$\{(t_s^i, t_e^i)\}_{i=1}^N, T \leftarrow f_p(\{x_s^i\}_{i=1}^N, x_g)$

$\mathbf{e}_g \leftarrow f_g(x_g) \in \mathbb{R}^{1\times d_g}$

$E_g \leftarrow \texttt{repeat}(\mathbf{e}_g, T) \in \mathbb{R}^{T\times d_g}$ // Broadcast the global prompt across all time frames.

Initialize $E_s \in \mathbb{R}^{T\times d_s}$ with zeros

for each (x_s^i, t_s^i, t_e^i) in segment prompts **do**

$j_s^i \leftarrow \lfloor t_s^i \cdot r/r_d \rfloor$, $i_e^i \leftarrow \lfloor t_e^i \cdot r/r_d \rfloor$

$\mathbf{e}_s^i \leftarrow f_s(x_s^i) \in \mathbb{R}^{1\times d_s}$

$E_s[j_s^i : j_e^i] \leftarrow \mathbf{e}_s^i$ // Broadcast each segment prompt to its corresponding temporal window.

end for

$E_{\text{cat}} \leftarrow \texttt{concat}(E_g, E_s, \text{dim} = -1)$, $E_{\text{cat}} \in \mathbb{R}^{T\times(d_g+d_s)}$

$E_{text} \leftarrow \texttt{out\_proj}(E_{\text{cat}})$

return fused embedding $E_{text} \in \mathbb{R}^{T\times d_{text}}$

with timestamps.

Below are the target global song description and lyrics. Please follow the instructions above and return the completed .lrc file directly.

Song Description
This pop rock ballad features a male vocalist delivering an emotional and uplifting melody. The mood is warm and introspective, with a gradually intensifying energy that enhances the song's heartfelt tone. The singer's voice is soulful and expressive, using subtle dynamic shifts to convey a sense of comfort and encouragement.

Lyrics
[This piece is the start of the song.]

[This piece is the intro of the song. The song segment features a spare and gentle piano motif, setting a contemplative and soothing mood ...]

[This piece is the first verse of the song. The segment features a tender vocal performance accompanied by a steady, melodic piano line ...]
Hey Jude, don't make it bad,
Take a sad song and make it better.
...

[This piece is the second verse of the song. The segment features a tender vocal performance accompanied by a steady, melodic piano line ...]
Hey Jude, don't be afraid,
You were made to go out and get her.
...

LRC Prediction:

C Workflow of Data Pipeline

Figure 3 shows the data curation pipeline, which comprises three key stages: (1) Quality filtering, where non-musical or low-quality clips are removed via metadata constraints and aesthetic scoring (using Audiobox Aesthetics and SongEval); (2) Lyrics processing, involving multilingual ASR transcription, and alignment validation against ground-truth LRC files via edit distance; and (3) Hierarchical music caption generation, where global and segment-level prompts are generated by Audio Flamingo 3, augmented with structural labels for controllability, and boundary markers are inserted at the start/end of each sample.

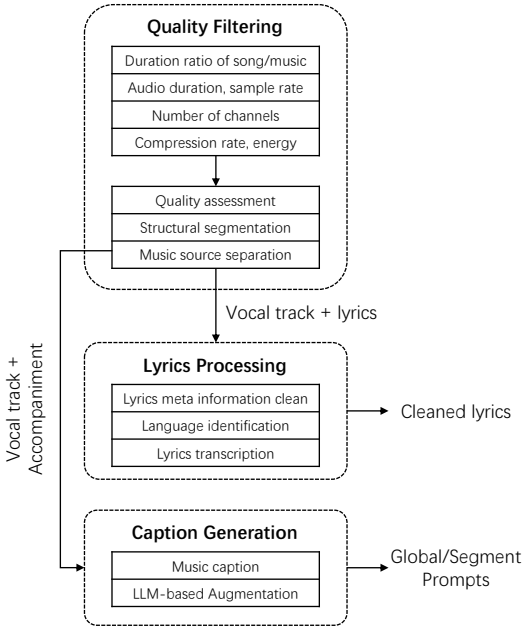

Figure 3: Overview of the data pipeline of SegTune.

D Audio-Flamingo 3 Caption Prompt

Prompt for Global Caption Generation

You are a helpful AI assistant. You need to act as a caption generator for music and generate descripitons in MusicCaps style. Describe the music in vivid detail, using the following rules:

1. Describe the details about genre, mood, feeling, ambience, and other notable features of the music.
2. Describe the singer's vocal characteristics, including gender, age range, vocal timbre, pitch range, and other notable features of the singer.
3. Keep the descripiton within 1-4 sentences.
4. Only provide details you are confident about. It is not compulsory to provide all details, but do not hallucinate.

Prompt for Segment Caption Generation

You are a helpful AI assistant. Describe the song segment as part of a complete piece of song in vivid detail according to what you hear. Generate the descripiton using the following rules:

1. Include the instrumentation, rhythm and melody style, mood, emotional's impact, intensity and change.
2. Mention any notable singing and playing techniques that occur and dynamic changes of the song.
3. Keep the descripiton within 1-3 sentences.
4. Only provide details you are confident about. It is not compulsory to provide all details, but do not hallucinate.

E Singer Gender Control Experiment and Visualization

To support a more interpretable and evidence-based comparison of the capacity of MuQ-MuLan and Qwen3-Embedding-0.6B in controlling vocal characteristics in song generation, we conduct controlled experiments and visualizations using *singer gender* as a representative attribute.

Specifically, we randomly sample 1,000 global prompts from the training dataset, each containing textual descriptions across multiple dimensions—including singer gender, musical style, and emotional tone. By applying rule-based string replacement, we invert the specified singer gender in each prompt (i.e., female vs. male), thereby constructing a paired control dataset of 2,000 prompts. We then extract text embeddings for all 2,000 prompts using both MuQ-MuLan and Qwen3-Embedding, respectively. Each set of embeddings is first reduced to 50 dimensions via Principal Component Analysis and subsequently visualized using t-distributed Stochastic Neighbor Embedding (t-SNE).

As shown in the figure 4 and figure 5, the t-SNE visualization reveals a stark contrast: embeddings from MuQ-MuLan exhibit near-complete overlap between female and male prompts in the embedding space, indicating an inability to encode gender-related vocal attributes, and the cosine distance of cluster centers for female and male groups is 0.002. In contrast, Qwen3-Embedding yields clearly separated clusters for the two genders, and the cosine distance of cluster centers for female and male groups is 0.107, providing direct empirical evidence of its superior discriminative capability.

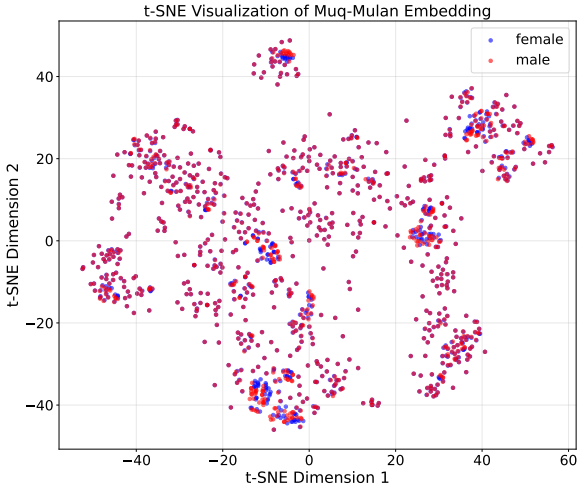

Figure 4: t-SNE visualization of Muq-Mulan embeddings on singer gender control.

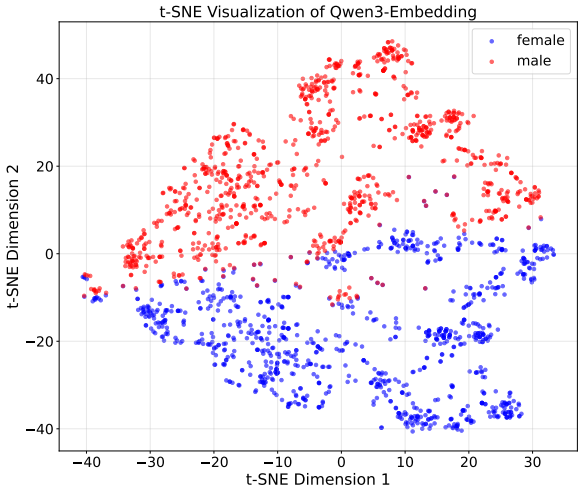

Figure 5: t-SNE visualization of Qwen3-Embedding on singer gender control.