

MUSTBENCH: Benchmarking and Advancing Temporal Grounding in Music LLMs

Daeyong Kwon^{1,2*}, Qiyu Wu^{2†}, Shinobu Kuriya², Junghyun Koo³, Shuyang Cui²,
Zhi Zhong², Wei-Hsiang Liao³, Hiromi Wakaki², Yuki Mitsufuji^{2,3}

¹Seoul National University

²Sony Group Corporation ³Sony AI

¹daeyongkwon@snu.ac.kr ^{2,3}{first_name.last_name}@sony.com

Abstract

Recent Large Audio-Language Models (LALMs) have demonstrated promising abilities in understanding musical content. However, whether their responses are grounded in the correct temporal regions of the audio remains underexplored. This limitation is particularly critical for music understanding, where key information often occurs as temporally localized events, such as instrument entries and rhythmic transitions. To address this gap, we introduce MUSTBENCH, a music-expert-validated benchmark designed to evaluate temporal grounding in LALMs through five temporally grounded question-answering tasks. To further improve temporal grounding in existing models, we propose MUST, a novel four-stage temporal optimization recipe spanning music encoder adaptation, LLM adaptation, LLM supervised fine-tuning, and RL-based optimization. Experiments on MUSTBENCH show that existing LALMs struggle with precise temporal grounding, while MUST brings significant improvements over strong baselines. These results establish temporal grounding as a key missing capability in current LALMs and position MUSTBENCH as a challenging benchmark for future research in temporally grounded music understanding.¹

and Music Flamingo (Ghosh et al., 2025) can generate descriptions of musical tracks with rich attributes, suggesting that LALMs are increasingly capable of understanding complex musical content.

However, detailed music description does not necessarily indicate temporal grounding. Temporal grounding is particularly critical for music understanding, as key information often occurs through temporally localized or evolving events, such as instrument entries and shifts in timbre, harmony, or tempo (Paulus et al., 2010). Since LALMs are known to sometimes hallucinate (Cheng et al., 2026), a plausible description such as “*the track evolves with the entry of a guitar*” is not sufficient evidence that the model is grounded in the audio. Such a claim remains difficult to verify unless the model can also identify the temporal region where the guitar enters. We refer to this capability as *music temporal grounding*: the ability to associate a textual claim about musical content with the specific time point or interval in the audio where the claim is acoustically supported. However, existing music benchmarks mainly evaluate track-level captioning (Agostinelli et al., 2023) or general audio question answering (QA) (Weck et al., 2024), leaving temporal grounding largely underexplored.

1 Introduction

Music understanding has long been a challenging problem due to the complex attributes in musical audio. Recent advances (Tang et al., 2023; Chu et al., 2024) in Large Audio-Language Models (LALMs) have extended their capabilities to non-speech domains such as music. State-of-the-art models such as Qwen3 Omni (Yang et al., 2025)

To address this gap, we introduce **Music Temporal Benchmark (MUSTBENCH)**, a benchmark validated by music experts for evaluating temporal grounding in LALMs. MUSTBENCH comprises five temporally grounded QA tasks: temporal source grounding (TSG), local transition recognition (LTR), transition-aware description (TAD), global temporal ordering (GTO), and mood trajectory reasoning (MTR). Together, these tasks assess a model’s ability to ground musical events to the specific moments or intervals at which they occur. Figure 1 presents examples from each task category and provides a preview of model performance. Our evaluation shows that existing models remain lim-

*Work done during an internship at Sony Group Corporation.

†Corresponding author: Qiyu Wu, qiyu.wu@sony.com

¹Code and benchmark data is available at <https://github.com/sony/MusTBench>

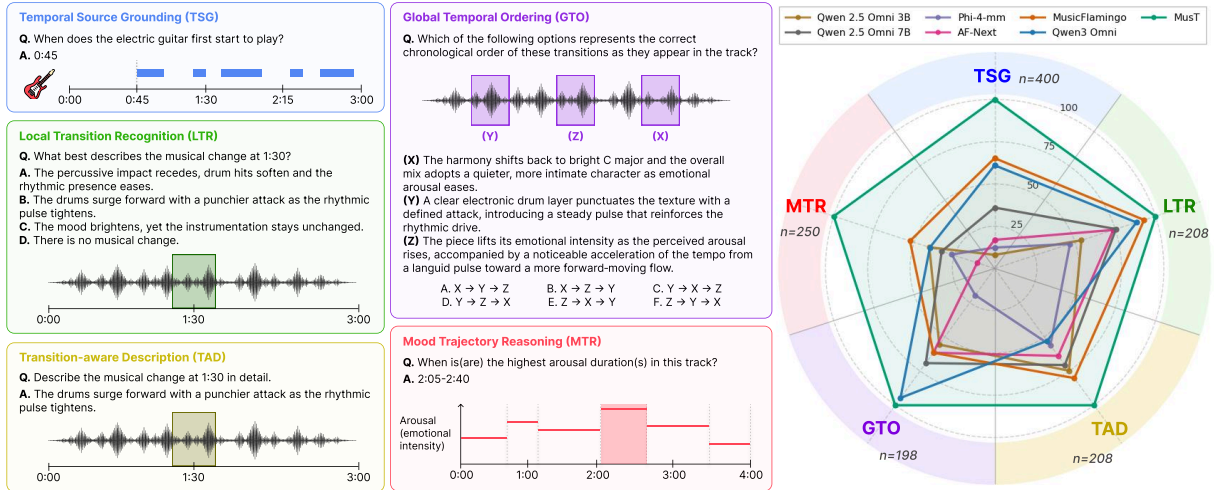


Figure 1: **(Left)** MUSTBENCH examples illustrating five types of temporally grounded music reasoning questions. **(Right)** Performance on MUSTBENCH comparing open-source baselines with MUST across five temporal grounding tasks, showing consistent gains from our approach. Values are normalized.

ited in temporal grounding. For example, in TSG, the most basic grounding task, a model is given a full music track and asked to predict when an instrument or vocal first enters or finally exits. Even on this straightforward task, current LALMs exhibit systematic failures. They struggle particularly with identifying offsets, often collapse their predictions to coarse temporal anchors such as 60s or 120s, and sometimes generate timestamps that fall outside the valid duration of the audio. Together with broader diagnostic on MUSTBENCH, these observations suggest that current models often rely on rough temporal priors rather than precisely grounding their responses in the input audio, indicating a lack of basic temporal perception ability.

To further improve temporal grounding in existing models, we propose MUST, a four-stage temporal optimization recipe spanning music encoder adaptation, LLM adaptation with timestamped music captions, supervised temporal QA fine-tuning, and RL-based optimization. This recipe equips the model with transition-aware temporal representations, adapts the LLM to timestamped music understanding, and directly optimizes timestamp- and interval-level grounding ability. Comprehensive experiments demonstrate that MUST brings significant improvements over strong baselines.

Our contributions are summarized as follows: (1) We identify temporal grounding as a key missing capability in current LALMs for music understanding; (2) We introduce MUSTBENCH, a music-expert-validated benchmark for evaluating temporal grounding in full-length music through five temporally grounded QA tasks; and (3) We

propose MUST, a four-stage temporal optimization recipe that significantly improves temporal grounding over strong LALM baselines. Together, these contributions provide a challenging benchmark and a practical training recipe for advancing temporal grounding in music understanding.

2 Related Work

Music LLMs. Recent LALMs have expanded music understanding from fixed-label prediction to captioning, question answering, and instruction following (Agostinelli et al., 2023; Deng et al., 2024). Music-oriented LLMs such as MU-LLaMA (Liu et al., 2024a) and MuMu-LLaMA (Liu et al., 2024b) align music representations with language models, while general audio-language models such as the Qwen-Omni series (Xu et al., 2025; Yang et al., 2025), and Music Flamingo (Ghosh et al., 2025) have shown strong performance on music reasoning tasks. These models can generate descriptions of musical attributes such as mood, genre, and tempo, while the temporal grounding ability of them remains underexplored.

Music Understanding Benchmarks. Existing audio-based music benchmarks primarily evaluate global music understanding. Datasets such as MusicQA (Liu et al., 2024a), MusicInstruct (Deng et al., 2024), MusicCaps (Agostinelli et al., 2023), MagnaTagATune (Law et al., 2009), and MuChoMusic (Weck et al., 2024) test whether models can recognize musical content or answer questions about the overall audio. While useful for evaluating music perception and language generation, these

benchmarks rarely require models to ground an answer to a specific timestamp or temporal interval.

Temporal Grounding. Temporal grounding has been studied in video, audio-visual, and general audio understanding, where models localize moments or sound events from natural language queries (Li et al., 2022; Chowdhury et al., 2024; Xu et al., 2021). Recent studies show that audio-language models can exhibit temporal biases, including hallucinated events and incorrect chronological ordering (Yao et al., 2025), and timestamp-aware audio captioning can improve temporal alignment (Kumar et al., 2026). However, music poses a distinct challenge, as meaningful changes often emerge through evolving instrumentation, rhythm, harmony, and emotional intensity, rather than appearing as isolated discrete sound events. In this work, we fill this gap by evaluating whether LALMs can ground such music-specific changes to concrete moments in the audio, and also provide a practical training recipe to improve existing models.

3 MUSTBENCH

We introduce MUSTBENCH: a benchmark for evaluating temporal grounding in music understanding. The construction of MUSTBENCH consists of (1) building a timestamped music caption dataset, and (2) generating five types of QA tasks for different aspects of temporal grounding. Figure 2 provides an overview of the full data generation pipeline.

3.1 Timestamped Music Caption Generation

To construct time-aware features, we first build timestamped captions from full-track audios in MTG-Jamendo (Bogdanov et al., 2019). These captions serve as intermediate annotations for identifying musical states and transitions. As shown in Figure 2-A, the process consists of four steps:

1. **Segmentation.** We apply a structural music segmentation model (Hao et al., 2025) to obtain segment boundaries, which serve as candidate transition timestamps.
2. **Mood-change modeling.** We collect human annotations for sampled transition clips and train a MERT-based predictor (Li et al., 2024) to estimate changes in emotional intensity around segment boundaries.
3. **Feature-grounded captioning.** We extract diverse musical features and use them to generate segment-level static captions and boundary-level dynamic captions.

Statistic	Overall	TSG	LTR	TAD	GTO	MTR
# QA pairs	1,264	400	208	208	198	250
# Unique Songs	517	246	208	208	198	250
Avg. duration	3m 42s	3m 51s	3m 31s	3m 31s	3m 26s	3m 18s

Table 1: Statistics of MUSTBENCH. The overall song count is computed after merging overlaps across task types. Audio duration is computed over unique songs.

QA Task	Train	Val	Test	Total
Temporal Source Grounding (TSG)	8,000	378	400	8,778
Local Transition Recognition (LTR)	8,000	764	208	8,972
Transition-Aware Description (TAD)	8,000	683	208	8,891
Global Temporal Ordering (GTO)	8,000	767	198	8,965
Mood Trajectory Reasoning (MTR)	8,000	1,975	250	10,225
Total	40,000	4,567	1,264	45,831

Table 2: Task-wise train/validation/test split statistics.

4. **Cross-validation and rewriting.** We verify instrument and key descriptions using additional music classifiers, then rewrite neighboring captions to improve coherence while preserving timestamps and musical facts.

The resulting timestamped captions provide temporally aligned musical evidence, used for later timestamped pretraining and QA generation. Implementation details are provided in §A.2.

3.2 MUSTBENCH QA Generation

To comprehensively evaluate temporal grounding ability, we propose five types of tasks as follows,

Temporal Source Grounding. TSG measures the most fundamental temporal grounding ability by asking models to identify when a specific sound source begins or ends. For instrument questions, we extract onset and offset times from MIDI-aligned instrument tracks in Slakh2100 (Manilow et al., 2019). We construct vocal questions from MTG-Jamendo (Bogdanov et al., 2019) using a source separator (Rouard et al., 2023), and label vocal onset and offset times by volume filtering. For both sources, we retain only examples where the target source has sufficiently high volume and ask only about its first or final audible occurrence, making the queried source perceptually identifiable and the ground truth temporally unambiguous.

Local Transition Recognition. LTR evaluates whether a model can identify the description corresponding to a specified transition timestamp. Given the full audio and a timestamp, the model must select the description that best matches the musical transition occurring at that moment from multiple choices. To make this task more challenging, we

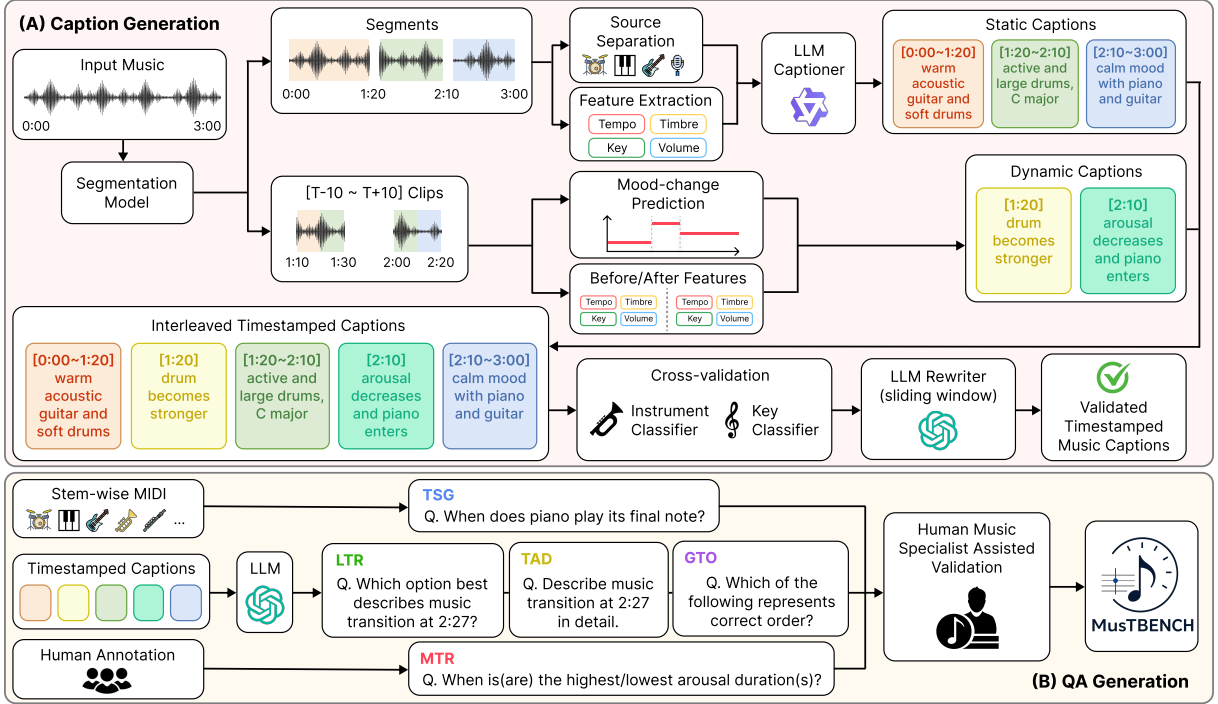


Figure 2: Overview of the MUSTBENCH construction pipeline. **(A)** Timestamped music captions are generated by segmentation, mood-change modeling, feature-grounded captioning, and cross-validation and rewriting. **(B)** QA pairs are generated from stem-wise MIDI annotations, timestamped music captions, and human annotations. All generated QA pairs are validated with assistance from human music experts to produce the final benchmark.

primarily use real transition descriptions from the same song but different timestamps as distractors, to prevent the shortcut where the model rely only on the overall musical context.

Transition-Aware Description. TAD evaluates open-ended transition understanding. Given the full audio and a transition timestamp, the model is asked to describe the musical change occurring around that moment. This removes answer-space constraints and tests whether the model can generate a correct temporally grounded description without relying on predefined options.

Global Temporal Ordering. GTO evaluates relative temporal reasoning over multiple transition events. We sample three transition descriptions from the same track and present them in random order. The model is then asked to determine their correct chronological order within the song, out of the six possible permutations. Unlike LTR and TAD which focus on local transition, this type assesses whether the model can reason over the global temporal structure of multiple events distributed throughout the track.

Mood Trajectory Reasoning. MTR evaluates whether models can localize the time intervals where the music reaches its highest or lowest emo-

tional intensity. Using transition boundaries from the segmentation model, we apply a trained mood-change predictor (details in §A.2.2) to label each transition clip by cumulatively summing change values, then identify the segments with the highest and lowest values as candidate answers. To ensure reliable supervision, we filter out cases with overly short segments or insufficient separation between the highest and lowest regions. Multiple answer spans are allowed when candidate segments have mood values within a predefined range.

Finally, we ask **music experts to review** the correctness, temporal alignment, and clarity of filtered QA pairs, and obtain 1,264 high-quality pairs. Details of human-assisted validation are in §A.3. Table 1 reports the statistics of MUSTBENCH. The noisy but larger-scale data is also used for training, detailed statistics are provided in Table 2.

3.3 Evaluation Metrics

Since MUSTBENCH evaluates various abilities, we design multiple task-specific metrics to measure the performance. For LTR and GTO, we report accuracy as they are multiple-choice tasks.

Temporal Source Grounding. For TSG, the model predicts a single timestamp. We evaluate whether the prediction falls within a tolerance

threshold T of the ground-truth timestamp:

$$\text{Hit@T} = \frac{1}{N} \sum_{i=1}^N \mathbb{I} \left(\left| t_{\text{pred}}^{(i)} - t_{\text{gt}}^{(i)} \right| \leq T \right).$$

Transition-Aware Description. For TAD, the model generates a free-form textual answer. We report METEOR (Banerjee and Lavie, 2005) for lexical overlap with the reference description and CLAP Score (Wu et al., 2023) for audio-text alignment. The CLAP Score is computed as the cosine similarity between the audio embedding of a 20-second transition clip and the text embedding of the generated description.

Mood Trajectory Reasoning. For MTR, a question may have multiple valid temporal durations. Let $\mathcal{P}$ and $\mathcal{G}$ denote the predicted and ground-truth duration sets, and let $U(\cdot)$ denote the union of durations in a set. We report Temporal IoU and F1, which measure the overlap between predicted and ground-truth temporal regions:

$$\text{IoU} = \frac{|U(\mathcal{P}) \cap U(\mathcal{G})|}{|U(\mathcal{P}) \cup U(\mathcal{G})|}, \quad \text{F1} = \frac{2|U(\mathcal{P}) \cap U(\mathcal{G})|}{|U(\mathcal{P})| + |U(\mathcal{G})|}$$

3.4 Performance of Existing Models

We evaluate representative open-source and closed-source LALMs, as listed in §A.5.2. Table 3 shows a clear gap between recognizing temporal information and precisely localizing it. Open-source models obtain relatively strong results on the multiple-choice tasks, LTR and GTO, where they only need to select among predefined options. However, their performance drops substantially on tasks that require direct temporal prediction. For example, several open-source models achieve competitive GTO accuracy, but their MTR scores remain consistently low, indicating difficulty in estimating continuous temporal boundaries. Closed-source models generally perform better on these localization-heavy tasks, but their results are still far from saturated. These findings suggest that current LALMs can often recognize temporally relevant musical changes, but precise temporal boundary estimation remains a major unresolved challenge. In addition, the results show that music temporal grounding is not simply determined by general model capability. Within the Gemini family, Flash variants often outperform their Pro counterparts, and Gemini 2.5 models obtain higher overall scores than Gemini 3 models. This suggests that temporal grounding is a specialized capability that depends on fine-grained audio-time alignment, rather than a direct byproduct of

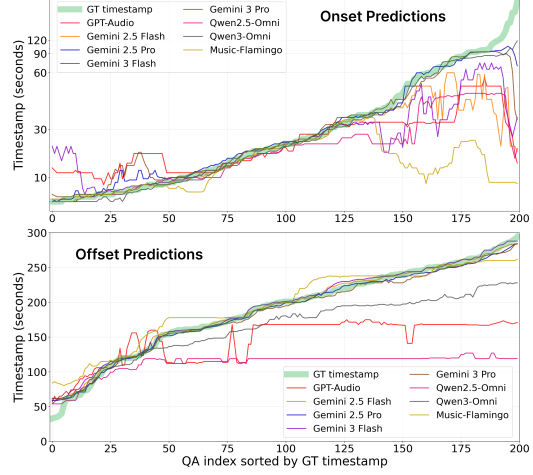


Figure 3: TSG predictions for onset and offset localization. To show overall trends, each line is smoothed using a sliding-window mean with a window size of 20.

stronger reasoning ability.

Qualitative Analysis on TSG. To better understand TSG behavior, we analyze TSG predictions in Figure 3. Overall, onset prediction is easier than offset prediction. Most models more reliably identify when a source first appears than when it disappears. For onset prediction, models are most accurate when the ground-truth timestamp falls around 10–30 seconds, while accuracy decreases for both early and later entries. Offset prediction shows heavier temporal instability. Models perform reasonably well within roughly the first 120 seconds, but predictions become increasingly noisy afterward, often collapsing to coarse duration anchors such as 120 and 180 seconds. These patterns suggest that some models rely on temporal priors rather than accurately grounding offsets in the input audio. Among the evaluated models, the Gemini series shows the strongest performance, while Qwen3 Omni remains competitive on onset prediction. We also observe out-of-range predictions beyond the input audio duration, indicating that current LALMs do not always respect input-specific temporal constraints. These limitations motivate our transition-aware dual-encoder model for fine-grained music temporal grounding. Full prediction details are provided in Figure 8.

4 MUST Training

The analysis on MUSTBENCH demonstrates that existing open-source models are still limited in temporally grounded music understanding. To further improve them, we propose MUST, a four-stage training recipe as illustrated in Figure 4. The

Model	Size	TSG			LTR	TAD			GTO	MTR			Total
		Onset Hit@3s	Offset Hit@3s	Avg.	Acc.	METEOR	CLAP Score	Avg.	Acc.	Temporal IoU	Temporal F1	Avg.	Avg.
Closed-source Models													
Gemini 2.5 Flash	–	60.0	71.5	65.8	56.7	11.2	34.5	22.9	42.4	24.8	33.1	29.0	41.8
Gemini 2.5 Pro	–	57.5	57.0	57.3	62.5	10.0	38.1	24.1	46.0	22.4	28.6	25.5	40.3
Gemini 3 Flash	–	55.0	42.5	48.8	51.4	10.7	37.9	24.3	47.0	25.7	34.2	30.0	38.1
Gemini 3 Pro	–	60.0	44.5	52.3	54.3	6.5	33.0	19.8	27.3	29.2	37.6	33.4	36.6
GPT Audio	–	21.5	1.5	11.5	42.3	12.0	33.0	22.5	36.9	13.8	19.6	16.7	22.6
GPT Audio 1.5	–	14.0	3.5	8.8	51.0	12.4	34.8	23.6	37.9	13.7	20.3	17.0	23.5
Open-source Models													
Phi-4-mm	6B	14.6	0.0	7.3	28.4	9.6	23.0	16.3	13.1	5.0	8.8	6.9	12.8
AF-Next	8B	17.0	3.0	10.0	44.7	9.0	29.4	19.2	41.4	2.2	3.4	2.8	18.8
Music Flamingo	8B	53.0	24.0	38.5	56.3	13.4	33.4	23.4	41.4	10.1	17.1	13.6	31.1
Qwen 2.5 Omni	3B	7.5	2.0	4.8	32.7	11.0	33.6	22.3	37.4	8.2	12.8	10.5	18.2
Qwen 2.5 Omni	7B	39.0	3.5	21.3	45.7	9.2	33.7	21.5	46.5	6.4	10.6	8.5	24.3
Qwen 3 Omni	30B-A3B	62.5	9.5	36.0	53.4	7.1	24.9	16.0	63.6	8.8	12.0	10.4	30.2
MuST (Ours)	3B	35.5 (+28.0)	41.0 (+39.0)	38.3 (+33.5)	58.2 (+25.5)	21.7 (+10.7)	35.4 (+1.8)	28.5 (+6.2)	57.1 (+19.7)	24.1 (+15.9)	31.4 (+18.6)	27.8 (+17.3)	38.1 (+19.9)
MuST (Ours)	7B	55.5 (+16.5)	62.5 (+59.0)	59.0 (+37.7)	60.6 (+14.9)	21.0 (+11.8)	34.6 (+0.9)	27.8 (+6.3)	67.2 (+20.7)	22.6 (+16.2)	29.1 (+18.5)	25.9 (+17.4)	44.1 (+19.8)

Table 3: Evaluation results on MUSTBENCH. The Avg. columns under TSG, TAD, and MTR denote the arithmetic mean of their corresponding sub-metrics. Numbers in parentheses indicate absolute improvements over the corresponding Qwen 2.5 Omni base model with the same model size.

LALM backbone is Qwen2.5 Omni (Xu et al., 2025) with a new proposed MUST encoder.

Stage 1: Transition-Aware Encoder Pretraining.

As we discussed above, existing LALMs can lack of basic temporal perception. Hence, we first adapt a MERT encoder (Li et al., 2024) with LoRA (Hu et al., 2022) to better capture temporally meaningful musical changes. The encoder is trained with two temporal objectives: transition probability prediction and mood-change prediction. For transition probability prediction, we use structural segment boundaries to construct Gaussian-smoothed boundary targets. Given the predicted transition logit z_t at frame t , the predicted probability $p_t = \sigma(z_t)$, and the Gaussian-smoothed target y_t , we optimize a BCE-Dice loss:

$$\mathcal{L}_{\text{trans}} = \text{BCE}(z, y) + 1 - \frac{2 \sum_t p_t y_t + s}{\sum_t p_t + \sum_t y_t + s}.$$

Here, s is a smoothing constant. The BCE term provides frame-level supervision, while the Dice term encourages overlap between predicted transition probabilities and sparse boundary regions. For mood-change prediction, we train the encoder to estimate changes in emotional intensity around transition boundaries using the annotations in §3.1. Given predicted and gold mood-change scores $\hat{a}$ and a , we use the concordance correlation coefficient loss (CCC loss):

$$\mathcal{L}_{\text{mood}} = 1 - \frac{2 \text{cov}(\hat{a}, a)}{\sigma_{\hat{a}}^2 + \sigma_a^2 + (\mu_{\hat{a}} - \mu_a)^2 + \epsilon}.$$

This objective encourages the predicted mood-change scores to match both the scale and relative variation of human annotations. The encoder is optimized using the sum of the transition and mood-change losses. The proposed MUST encoder pro-

vides fine-grained temporal representations for the following stages. Details are described in §A.4.1.

Stage 2: Timestamped Caption Pretraining.

Inspired by Touvron et al. (2023), we train the dual-encoder LLM on timestamped music captioning for modality alignment. MUST encoder provides fine-grained acoustic and temporal representations, while the frozen Qwen audio encoder provides semantic representations. A learnable projector maps MUST encoder outputs into the LLM embedding space, and sinusoidal time embeddings encode the absolute timestamp of each MUST token. These tokens are concatenated and fed into the LLM. To use the limited audio token budget more effectively, we propose *transition-aware dynamic sampling* based on the transition probabilities. Instead of uniformly sampling tokens over the full track, the model allocates more tokens to regions with high transition probability while preserving global coverage. The model is then trained to generate timestamped captions, learning both timestamp localization and segment-level music description. More details are described in §A.4.2.

Stage 3: QA Fine-Tuning.

Next, we fine-tune the model with LoRA on the five QA tasks on MUSTBENCH training set. This stage adapts the model from timestamped captioning to five temporal understanding tasks. Because the QA tasks have heterogeneous answer formats, we use an answer-only, sample-normalized, task-balanced objective so that long-form description tasks do not dominate training. The full objective is provided in §A.4.3.

Stage 4: GRPO Training.

Finally, we apply Group Relative Policy Optimization (GRPO) (Shao et al., 2024) to timestamp-centric tasks. Unlike token-level cross-entropy, GRPO allows us to opti-

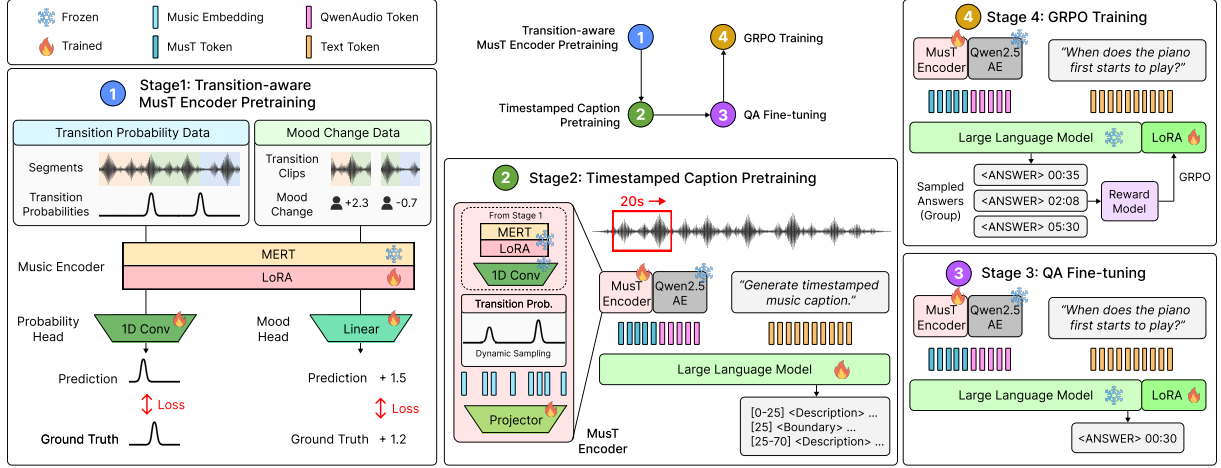


Figure 4: Overview of the proposed four-stage training pipeline and model architecture. **Stage 1: Transition-aware MUST encoder pretraining.** We train the encoder on transition probability and mood change prediction tasks. **Stage 2: Timestamped caption pretraining.** The trained MUST encoder extracts MUST tokens, which are combined with tokens from a frozen Qwen2.5-Omni audio encoder. The LLM is fine-tuned to generate timestamped music captions. **Stage 3: QA fine-tuning.** The LLM is further trained with LoRA to answer temporally grounded music questions. **Stage 4: GRPO training.** Reinforcement learning (GRPO) is applied to improve temporal grounding ability.

mize task-level rewards that directly reflect temporal grounding quality. We use a continuous reward so that predictions closer to the gold timestamps or intervals receive higher rewards, even when they do not exactly satisfy hard evaluation metrics.

For TSG, which requires a single timestamp prediction, we convert the absolute temporal error into an exponential reward:

$$r_{\text{TSG}} = \exp\left(-\frac{|\hat{t} - t^*|}{15}\right) - 0.5 \cdot \mathbf{1}_{\text{out}} - 1.0 \cdot \mathbf{1}_{\text{fmt}},$$

where $\hat{t}$ and t^* denote the predicted and gold timestamps. The 15-second scale controls the temporal tolerance of the reward, while $\mathbf{1}_{\text{out}}$ and $\mathbf{1}_{\text{fmt}}$ penalize out-of-range and invalid-format predictions.

For MTR, which requires temporal interval prediction, we use a Gaussian-smoothed soft-F1 reward. Given predicted intervals P and gold intervals G , we convert them into temporal masks and apply Gaussian smoothing to obtain P_{soft} and G_{soft} . The reward is based on:

$$\text{SoftF1}_{\text{gaussian}}(P, G) = \frac{2\langle P_{\text{soft}}, G_{\text{soft}} \rangle}{\|P_{\text{soft}}\|_2^2 + \|G_{\text{soft}}\|_2^2 + \epsilon},$$

with the final reward penalized for out-of-range intervals and invalid answer formats. This soft-F1 reward gives partial credit to near-miss interval predictions, encouraging the model to first locate the correct temporal region and then refine its boundaries. Full details are provided in §A.4.4.

5 Experiments

5.1 Experimental Setup

Baseline Models. We evaluate representative open-source and closed-source Large Audio-Language Models (LALMs) that support music or general audio understanding. For open-source baselines, we include Qwen 2.5 Omni (Xu et al., 2025), Qwen 3 Omni (Yang et al., 2025), MusicFlamingo (Ghosh et al., 2025), AudioFlamingo-Next (Ghosh et al., 2026), and Phi-4-mm (Abouelenin et al., 2025). For closed-source baselines, we evaluate GPT Audio and GPT Audio 1.5 (Hurst et al., 2024), Gemini 2.5 Pro/Flash (Comanici et al., 2025), and Gemini 3 Pro/Flash (Google, 2025). All baseline models are evaluated in a zero-shot setting to assess their inherent temporal grounding ability without task-specific adaptation.

Inference Protocol. For each benchmark example, we provide the model with the full music track and the corresponding task-specific instruction. No task-specific demonstrations or in-context examples are provided. All models are evaluated with temperature set to 0 to ensure deterministic inference. For multiple-choice tasks, we extract the selected option from the model response. For timestamp and interval prediction tasks, we parse temporal expressions from the generated text and treat invalid formats as incorrect.

Configuration					TSG		LTR	TAD		GTO	MTR		Total
Ablation	Setting	Caption	QA	GRPO	Onset Hit@3s	Offset Hit@3s	Acc.	METEOR	CLAP Score	Acc.	Temporal IoU	Temporal F1	Avg.
Training stage	Base model	✗	✗	✗	39.0	3.5	45.7	9.2	33.7	46.5	6.4	10.6	24.3
	Caption only	✓	✗	✗	40.5	6.0	50.5	8.6	27.6	28.8	16.3	23.0	25.2
	QA only	✗	✓	✗	30.0	49.5	53.4	20.5	35.3	61.1	19.1	25.5	36.8
	Caption + QA	✓	✓	✗	40.0	63.5	61.1	21.1	34.3	65.7	21.4	28.0	41.9
	Full	✓	✓	✓	55.5	62.5	60.6	21.0	34.6	67.2	22.6	29.1	44.1
MUST token rate	6.66 tokens/sec	✓	✓	✗	40.0	63.5	61.1	21.1	34.3	65.7	21.4	28.0	41.9
	3.33 tokens/sec	✓	✓	✗	35.0	63.5	63.5	20.7	32.9	65.7	22.2	29.0	41.6
	0 tokens/sec	✓	✓	✗	21.5	0.5	37.0	14.5	37.5	53.0	13.8	18.7	24.6
Sampling strategy	Uniform sampling	✓	✓	✗	40.0	60.5	55.8	20.5	32.6	65.2	18.9	24.7	39.8
	Dynamic sampling	✓	✓	✗	40.0 (+0.0)	63.5 (+3.0)	61.1 (+5.3)	21.1 (+0.6)	34.3 (+1.7)	65.7 (+0.5)	21.4 (+2.5)	28.0 (+3.3)	41.9 (+2.1)

Table 4: Ablation on training stages, MUST token rate, and sampling strategy. The 0-token setting removes the MUST tokens and relies only on the Qwen2.5-Omni audio encoder.

5.2 Results

In addition to the baseline diagnostic in §3.4, we evaluate our model on MUSTBENCH. Implementation details are provided in §A.5.2. Table 3 summarizes the overall performance of MUST. MUST 7B achieves the highest overall score (44.1), improving over the corresponding base model by 19.8 pp. It also consistently improves over the base model across all task types. The most substantial gain appears in offset localization, where MUST improves over the base model by 59.0 pp. This is particularly important because offset detection is one of the weakest aspects of baseline models. The 3B version also shows meaningful gains, improving the overall score over the corresponding base model by 19.9 pp. This indicates that our training pipeline is effective across model scales. MUST also achieves the highest performance on global temporal ordering, showing that the proposed training pipeline improves not only local timestamp prediction but also broader temporal reasoning over musical changes.

5.3 Ablation Studies

Multi-Stage Training. We analyze the contribution of each training stage in our pipeline, including timestamped caption pretraining, QA fine-tuning, and GRPO training. As shown in Table 4, caption pretraining alone yields only a modest improvement, increasing the overall score from 24.3 to 25.2. In contrast, QA fine-tuning alone substantially improves the score to 36.8, demonstrating that task-specific supervision is essential for temporally grounded music reasoning. Combining caption pretraining with QA fine-tuning further raises the overall score to 41.9, suggesting that timestamped caption pretraining provides useful temporal alignment and transition-aware initialization for downstream QA learning. Finally, adding GRPO achieves the best overall score of 44.1. Be-

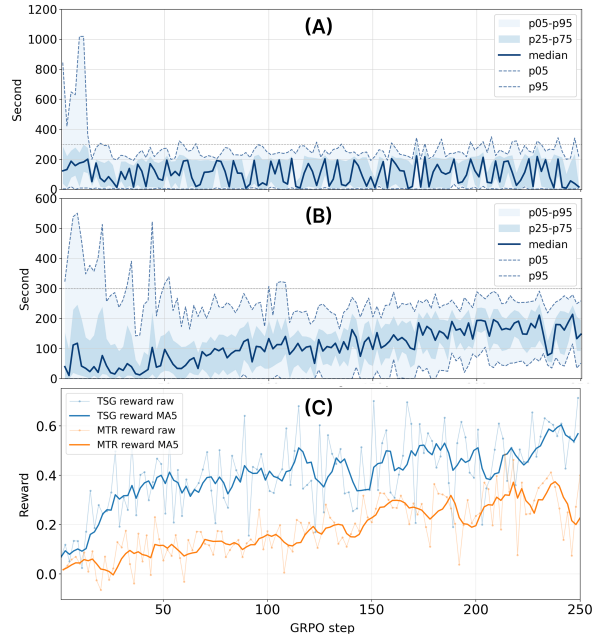


Figure 5: (A) TSG predictions during GRPO training. (B) MTR predictions during GRPO training. (C) TSG, MTR training reward curves.

yond this aggregate improvement, GRPO also improves the validity of timestamp predictions. Before GRPO, the model sometimes generates timestamps outside the valid input audio duration. As shown in Figure 5, these out-of-range predictions decrease as GRPO training progresses. This suggests that timestamp-aware preference optimization calibrates MUST toward temporally valid outputs, encouraging predictions that are not only close to the target timestamps or intervals but also constrained by the actual audio duration. Overall, these results show that the training stages play complementary roles, where caption pretraining supports temporal alignment, QA fine-tuning provides task-specific reasoning ability, and GRPO improves the temporal validity of timestamp-centric predictions.

MUST Tokens. To examine the contribution of the MUST tokens in our dual-encoder architecture, we

Encoder	Music Tagging ROC	Genre Cls. Acc.	Key Cls. F1
Qwen2Audio [†]	-	89.99	34.10
MERT	85.83	84.67	36.07
MUST	86.00	84.67	32.68

Table 5: Comparison between the encoders on three downstream music understanding tasks. Results marked with [†] are taken from Ghosh et al. (2025).

vary the token rate of the MUST encoder during caption pretraining and QA fine-tuning. We compare 3.33 and 6.66 tokens/sec, corresponding to maximum budgets of 1000 and 2000 tokens for a 5 minute audio. As shown in Table 4, removing these tokens substantially degrades performance, reducing the overall score from 41.9 to 24.6. Using 3.33 tokens/sec already achieves an overall score of 41.6, comparable to the 6.66 tokens/sec setting. Given that the original Qwen2.5-Omni audio encoder uses 25 tokens/sec, these results show that our MUST tokens provide substantial temporal grounding gains with only a small number of additional tokens.

Dynamic Sampling. To evaluate the contribution of our transition probability-guided dynamic sampling, we replace it with uniform sampling while keeping all other settings unchanged. As shown in Table 4, dynamic sampling improves the overall score from 39.8 to 41.9. The gains are particularly clear on tasks that require fine-grained access to musically important regions, where allocating more MUST tokens around transition points is likely to be beneficial. These results suggest that transition-aware dynamic sampling better captures temporally localized musical changes than uniform sampling.

Knowledge Retention in the MUST Encoder.

We evaluate whether MUST encoder training preserves the general musical representations of the original MERT encoder. For each encoder, we train only a task-specific linear head. As shown in Table 5, the MUST encoder maintains comparable performance to MERT on music tagging and genre classification, although key classification slightly decreases, which suggests that MUST encoder largely preserves general music understanding while specialized for temporal grounding.

6 Conclusion

We introduced MUSTBENCH, a benchmark for evaluating temporal grounding of LALMs. Through five temporal grounding tasks, MUSTBENCH reveals that existing LALMs still lack reliable temporal grounding ability in music under-

standing. To address this limitation, we further proposed MUST, a novel four-stage temporal optimization recipe including music encoder and LLM tuning. Comprehensive experiments and ablation study show that MUST brings significant improvements on strong baselines. Overall, MUSTBENCH establishes temporal grounding as a key challenge for future research and provides a foundation for developing models that understand not only what happens in music, but also when it happens.

Limitations

This work has two main limitations. First, MUSTBENCH relies on an automatic data generation pipeline, so residual noise may remain despite multiple validation steps. Structural boundaries may miss musically meaningful transitions, and automatic components may introduce subtle errors. Second, mood trajectory reasoning is subjective, as perceived arousal can vary across listeners, genres, and contexts. Thus, the arousal intervals in MUSTBENCH should be viewed as benchmark-specific annotations rather than definitive judgments of musical emotion.

Ethical Considerations

MUSTBENCH is intended for research on temporally grounded music understanding. Since it is derived from existing music datasets and annotations, all released annotations, metadata, evaluation scripts, and processing code should comply with the licenses and access conditions of the original datasets, models, and tools; restricted audio content will not be redistributed. The benchmark focuses on musical events and affective changes rather than personal identity, but recordings with vocals should not be used to identify singers or infer sensitive attributes. Model outputs may contain incorrect timestamps or descriptions and should not be used as authoritative evidence in copyright, forensic, or commercial rights settings without expert verification. MUSTBENCH may also inherit dataset and tool biases across genres, instruments, languages, and cultural traditions. AI-based writing assistance was used only for language polishing and clarity, while all research ideas, design decisions, experiments, analyses, and conclusions were developed and verified by the authors.

References

- Abdelrahman Abouelenin, Atabak Ashfaq, Adam Atkinson, Hany Awadalla, Nguyen Bach, Jianmin Bao, Alon Benhaim, Martin Cai, Vishrav Chaudhary, Congcong Chen, and 1 others. 2025. Phi-4-mini technical report: Compact yet powerful multimodal language models via mixture-of-loras. *arXiv preprint arXiv:2503.01743*.
- Sandhini Agarwal, Lama Ahmad, Jason Ai, Sam Altman, Andy Applebaum, Edwin Arbus, Rahul K Arora, Yu Bai, Bowen Baker, Haiming Bao, and 1 others. 2025. gpt-oss-120b & gpt-oss-20b model card. *arXiv preprint arXiv:2508.10925*.
- Andrea Agostinelli, Timo I Denk, Zalan Borsos, Jesse Engel, Mauro Verzetti, Antoine Caillon, Qingqing Huang, Aren Jansen, Adam Roberts, Marco Tagliasacchi, and 1 others. 2023. Musiclm: Generating music from text. *arXiv preprint arXiv:2301.11325*.
- Satanjeev Banerjee and Alon Lavie. 2005. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72.
- Sebastian Böck, Filip Korzeniowski, Jan Schlüter, Florian Krebs, and Gerhard Widmer. 2016. Madmom: A new python audio and music signal processing library. In *Proceedings of the 24th ACM international conference on Multimedia*, pages 1174–1178.
- Dmitry Bogdanov, Nicolas Wack, Emilia Gómez Gutiérrez, Sankalp Gulati, Perfecto Herrera Boyer, Oscar Mayor, Gerard Roma Trepas, Justin Salamon, José Ricardo Zapata González, and Xavier Serra. 2013. Essentia: An audio analysis library for music information retrieval. In *Britto A, Gouyon F, Dixon S, editors. 14th Conference of the International Society for Music Information Retrieval (ISMIR); 2013 Nov 4-8; Curitiba, Brazil.: ISMIR; 2013. p. 493-8*. International Society for Music Information Retrieval (ISMIR).
- Dmitry Bogdanov, Minz Won, Philip Tovstogan, Alastair Porter, and Xavier Serra. 2019. The mtg-jamendo dataset for automatic music tagging. ICML.
- Xize Cheng, Dongjie Fu, Chenyuhao Wen, Shannon Yu, Zehan Wang, Shengpeng Ji, Siddhant Arora, Tao Jin, Shinji Watanabe, and Zhou Zhao. 2026. Aha-bench: Benchmarking audio hallucinations in large audio-language models. *Advances in Neural Information Processing Systems*, 38.
- Sanjoy Chowdhury, Sayan Nag, Subhrajyoti Dasgupta, Jun Chen, Mohamed Elhoseiny, Ruohan Gao, and Dinesh Manocha. 2024. Meerkat: Audio-visual large language model for grounding in space and time. In *European Conference on Computer Vision*, pages 52–70. Springer.
- Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhifang Guo, Yichong Leng, Yuanjun Lv, Jinzheng He, Junyang Lin, and 1 others. 2024. Qwen2-audio technical report. *arXiv preprint arXiv:2407.10759*.
- Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, and 1 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.
- Zihao Deng, Yinghao Ma, Yudong Liu, Rongchen Guo, Ge Zhang, Wenhui Chen, Wenhao Huang, and Emmanouil Benetos. 2024. Musilingo: Bridging music and text with pre-trained language models for music captioning and query response. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 3643–3655.
- Sreyan Ghosh, Arushi Goel, Kaousheik Jayakumar, Lasha Koroshinadze, Nishit Anand, Zhifeng Kong, Siddharth Gururani, Sang-gil Lee, Jaehyeon Kim, Aya Aljafari, and 1 others. 2026. Audio flamingo next: Next-generation open audio-language models for speech, sound, and music. *arXiv preprint arXiv:2604.10905*.
- Sreyan Ghosh, Arushi Goel, Lasha Koroshinadze, Sang-gil Lee, Zhifeng Kong, Joao Felipe Santos, Ramani Duraiswami, Dinesh Manocha, Wei Ping, Mohammad Shoeybi, and 1 others. 2025. Music flamingo: Scaling music understanding in audio language models. *arXiv preprint arXiv:2511.10289*.
- Google. 2025. A new era of intelligence with gemini 3. <https://blog.google/products-and-platforms/products/gemini/gemini-3/>. Official Google announcement. Accessed: 2026-05-08.
- Chunbo Hao, Ruibin Yuan, Jixun Yao, Qixin Deng, Xinyi Bai, Wei Xue, and Lei Xie. 2025. Songformer: Scaling music structure analysis with heterogeneous supervision. *arXiv preprint arXiv:2510.02797*.
- Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Liang Wang, Weizhu Chen, and 1 others. 2022. Lora: Low-rank adaptation of large language models. *Iclr*, 1(2):3.
- Eric Humphrey, Simon Durand, and Brian McFee. 2018. Openmic-2018: An open data-set for multiple instrument recognition. In *ISMIR*, pages 438–444.
- Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.
- Sonal Kumar, Prem Seetharaman, Ke Chen, Oriol Nieto, Jiaqi Su, Zhepei Wang, Rithesh Kumar, Dinesh Manocha, Nicholas J Bryan, Zeyu Jin, and 1 others. 2026. Tac: Timestamped audio captioning. *arXiv preprint arXiv:2602.15766*.

- Edith Law, Kris West, Michael I Mandel, Mert Bay, and J Stephen Downie. 2009. Evaluation of algorithms using games: The case of music tagging. In *ISMIR*, pages 387–392.
- Guangyao Li, Yake Wei, Yapeng Tian, Chenliang Xu, Ji-Rong Wen, and Di Hu. 2022. Learning to answer questions in dynamic audio-visual scenarios. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 19108–19118.
- Yizhi Li, Ruibin Yuan, Ge Zhang, Yinghao Ma, Xin-gran Chen, Hanzhi Yin, Chenghao Xiao, Chenchua Lin, Anton Ragni, Emmanouil Benetos, and 1 others. 2024. Mert: Acoustic music understanding model with large-scale self-supervised training. In *International Conference on Learning Representations*, volume 2024, pages 12181–12204.
- Shansong Liu, Atin Sakkeer Hussain, Chenshuo Sun, and Ying Shan. 2024a. Music understanding llama: Advancing text-to-music generation with question answering and captioning. In *ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 286–290. IEEE.
- Shansong Liu, Atin Sakkeer Hussain, Qilong Wu, Chenshuo Sun, and Ying Shan. 2024b. Mumu-llama: Multi-modal music understanding and generation via large language models. *arXiv preprint arXiv:2412.06660*, 3(5):6.
- Ilya Loshchilov and Frank Hutter. 2016. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*.
- Ilya Loshchilov and Frank Hutter. 2017. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*.
- Ethan Manilow, Gordon Wichern, Prem Seetharaman, and Jonathan Le Roux. 2019. Cutting music source separation some Slakh: A dataset to study the impact of training data quality and quantity. In *Proc. IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*. IEEE.
- Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, Oriol Nieto, and 1 others. 2015. librosa: Audio and music signal analysis in python. *SciPy*, 2015(18-24):7.
- Jouni Paulus, Meinard Müller, and Anssi Klapuri. 2010. State of the art report: Audio-based music structure analysis. In *Ismir*, pages 625–636. Utrecht.
- Simon Rouard, Francisco Massa, and Alexandre Défossez. 2023. Hybrid transformers for music source separation. In *ICASSP 2023-2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 1–5. IEEE.
- Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang, Mingchuan Zhang, YK Li, Yang Wu, and 1 others. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *arXiv preprint arXiv:2402.03300*.
- Changli Tang, Wenyi Yu, Guangzhi Sun, Xianzhao Chen, Tian Tan, Wei Li, Lu Lu, Zejun Ma, and Chao Zhang. 2023. Salmonn: Towards generic hearing abilities for large language models. *arXiv preprint arXiv:2310.13289*.
- Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, and 1 others. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.
- Benno Weck, Iliaria Manco, Emmanouil Benetos, Elio Quinton, George Fazekas, and Dmitry Bogdanov. 2024. Muchomusic: Evaluating music understanding in multimodal audio-language models. *arXiv preprint arXiv:2408.01337*.
- 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 *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP*.
- Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang Fan, Kai Dang, and 1 others. 2025. Qwen2.5-omni technical report. *arXiv preprint arXiv:2503.20215*.
- Xuenan Xu, Heinrich Dinkel, Mengyue Wu, and Kai Yu. 2021. Text-to-audio grounding: Building correspondence between captions and sound events. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 606–610. IEEE.
- An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.
- Jiayu Yao, Shenghua Liu, Yiwei Wang, Rundong Cheng, Lingrui Mei, Baolong Bi, Zhen Xiong, and Xueqi Cheng. 2025. Not in sync: Unveiling temporal bias in audio chat models. *arXiv preprint arXiv:2510.12185*.

A Appendix

A.1 Overview

This appendix provides additional details on benchmark construction, model training, evaluation protocols, and supplementary analyses.

- **Timestamped music caption construction.** (§A.2) We describe the construction pipeline for source audio filtering and transition clip extraction, mood-change annotation, feature-grounded captioning, caption cross-validation.

- **Human-assisted Validation and Annotator Details.** (§A.3) We provide details on the human-assisted validation process used to finalize the QA dataset, as well as information about the human annotators who participated in this work.
- **Model training.** (§A.4) We provide implementation details for transition-aware encoder pretraining, timestamped caption pretraining, supervised QA fine-tuning, and GRPO reward design.
- **Additional analyses and experimental details.** (§A.5) We include supplementary ablations, qualitative examples, dataset statistics, feature statistics, baseline inference protocols, and training hyperparameters.

A.2 Timestamped Caption Generation Details

This section provides additional details on the construction of timestamped music caption generation.

A.2.1 Source Audio Filtering and Transition Clip Extraction

Source Audio Filtering. We use MTG-Jamendo (Bogdanov et al., 2019) as the source audio collection, which contains 55K full-track music recordings. From this collection, we retain 41K tracks with durations between 1 and 5 minutes. This duration range is chosen because the tracks are long enough to contain multiple musical events and structural changes, while remaining computationally tractable for dense segmentation, caption generation, and downstream QA construction.

Structural Segmentation and Transition Clip Extraction. To obtain candidate timestamps for musically meaningful changes, we apply a structural music segmentation model (Hao et al., 2025) to each track. The resulting segment boundaries are treated as transition timestamps. For each boundary at timestamp T , we extract a 20-second transition clip spanning $[T - 10, T + 10]$, where the transition point is centered in the clip. These transition clips are used for mood-change annotation, mood-change prediction, and dynamic caption generation. Applying this segmentation procedure yields approximately 120K transition clips.

Sampling Transition Clips for Human Annotation. Since annotating all transition clips is impractical, we construct a representative subset for human annotation. For each transition clip, we separately encode the pre-boundary segment $[T - 10, T]$ and the post-boundary segment

$[T, T + 10]$ using MERT (Li et al., 2024). We then cluster the resulting transition embeddings and uniformly sample 1,000 clips across clusters. This procedure encourages diversity in the annotated subset by covering different types of musical transitions rather than over-sampling frequent transition patterns.

A.2.2 Mood-Change Annotation and Predictor Training Details

Annotation Protocol. For the selected 1,000 transition clips, we collect human annotations of mood change from a pool of five annotators. Each clip is labeled by three annotators. Annotators listen to the 20-second transition clip with the transition point centered and rate the change in emotional intensity across the boundary on a discrete scale from -3 to 3 . Negative values indicate that the music becomes less intense after the transition, positive values indicate that it becomes more intense, and zero indicates little or no perceived arousal change.

Quality Control. To ensure annotation reliability, we remove clips with low inter-annotator agreement. Specifically, we discard 200 clips whose annotations show weak agreement among annotators, resulting in 800 high-quality transition samples. The retained annotations show a correlation of 0.88, indicating that the filtered labels provide reliable supervision for modeling arousal changes around musical transitions.

Annotator-Aware Mood-Change Prediction. Because perceived musical arousal can be subjective, we explicitly model annotator-specific variation. We attach five regression heads to a MERT encoder, one for each annotator, and train each head using only the samples labeled by the corresponding annotator. At inference time, we average the predictions from the five heads to obtain a robust estimate of the mood-change score. The resulting predictor is applied to transition clips throughout the dataset and used as an affective cue for dynamic caption generation.

A.2.3 Feature-Grounded Captioning Details

Feature Extraction. For each structural segment, we extract musical features that can provide grounded evidence for caption generation. These include tempo, key, and volume-related features computed using standard audio analysis libraries, including librosa (McFee et al., 2015), mad-

mom (Böck et al., 2016), and Essentia (Bogdanov et al., 2013). These features are used to constrain the captioning process so that generated descriptions reflect measurable properties of the audio rather than relying only on free-form model priors. The extracted feature set is summarized in Table 10.

In addition to global musical features, we apply a source separation model (Rouard et al., 2023) to obtain stem-level signals. The separation model provides coarse 4-stem outputs (vocals, drums, bass, and others). These source-level cues provide evidence about the presence and activity of major musical sources within each segment, which helps reduce hallucinations in instrument- and vocal-related descriptions.

Static Caption Generation. For each segment, we provide the extracted musical features and source-level cues as grounded inputs to the caption generation model. The model generates a segment-level static caption that describes the musical state within the segment, including instrumentation, rhythm, harmony, texture, and mood when supported by the extracted features. These static captions are aligned with segment-level timestamp intervals and are intended to describe what is present within each segment.

Dynamic Caption Generation. For each segment boundary, we compare the features extracted from the pre-boundary and post-boundary segments. We also include the predicted mood-change score from the mood-change predictor. Using these before/after features, the caption generation model produces a dynamic caption describing how the music changes across the transition. These dynamic captions capture localized musical changes such as instrument entries or exits, rhythmic intensification, texture reduction, harmonic shifts, and changes in emotional intensity.

Interleaving Static and Dynamic Captions. We combine static and dynamic captions into a single timestamped caption sequence. Static captions are associated with segment intervals, while dynamic captions are associated with transition timestamps. This produces an interleaved description of the full track, where relatively stable musical states and local transition events are both explicitly grounded in time.

Sliding-Window Rewriting. Because static and dynamic captions are initially generated indepen-

dently, the resulting caption sequence can lack coherence when read as a full-track description. To address this issue, we apply a sliding-window rewriting step using gpt-oss-120b (Agarwal et al., 2025). The rewriting model sequentially refines neighboring captions so that the final timestamped caption reads as a coherent description of the full track. During rewriting, the model is instructed to preserve the original musical facts, timestamps, and temporal structure, while improving fluency and local coherence. Examples before and after rewriting are shown in Figure 6 and Figure 7.

A.2.4 Caption Cross-Validation Details

An initial human validation study revealed that some generated captions contained errors in instrument and key descriptions. For instrument descriptions, the source separation model provides only coarse 4-stem outputs, which are insufficient for distinguishing fine-grained instruments such as guitar or piano. We therefore train a MERT-based classifier on OpenMIC-2018 (Humphrey et al., 2018) to classify the audio into nine fine-grained instrument categories. The classifier is used to verify instrument mentions in generated captions, and unsupported descriptions are revised before QA generation. For key descriptions, we apply a confidence-based verification rule. A key mention is retained when the classifier assigns a probability of at least 0.9 to the mentioned key, and removed when the probability is below 0.1. For intermediate confidence scores, we keep the original caption unchanged. Tables 8 and 9 show statistics of instruments in the caption dataset.

A.3 Human-assisted Validation and Annotator Details

Human-assisted Validation We apply task-specific validation procedures to ensure the reliability of the final benchmark annotations. Detailed instructions are included in Figures 9, 10, and 11.

For TSG, we do not conduct additional human validation. The ground-truth timestamps are obtained from MIDI-aligned instrument tracks and separated vocal tracks. We further retain only examples where the target source is clearly audible according to a volume-based filtering criterion. Since the resulting onset and offset timestamps are derived from structured source-level annotations and filtered for perceptual salience, additional human correction is not required.

For LTR and TAD, we provide annotators with

the full audio, the transition timestamp T , and the corresponding transition description. Annotators are asked to verify whether the description accurately matches the musical change occurring around T . If the description contains incorrect or unsupported musical content, they are instructed to revise it so that it correctly reflects the audio at the given timestamp.

For GTO, we provide annotators with three transition descriptions (X, Y, Z) along with their corresponding timestamps. Annotators first verify whether each description is temporally aligned with the actual musical change in the audio. If any description is inaccurate or misaligned, they are instructed to correct it before the chronological ordering question is finalized. This ensures that the task evaluates temporal ordering rather than errors in the transition descriptions themselves.

For MTR, mood trajectory reasoning, music experts directly listen to the audio and annotate the temporal regions with the highest or lowest arousal, depending on the question. Annotators are allowed to provide up to four valid duration intervals when multiple regions exhibit comparable arousal levels. These expert annotations are used as the ground-truth intervals for evaluating mood trajectory reasoning.

Human Annotation. Human annotation and validation were conducted by temporary research support staff affiliated with our institution. The annotators were not recruited through a public crowdsourcing platform, and their duties were not limited to this particular annotation assignment. Their employment and compensation complied with Japanese labor laws and institutional procedures. Annotators were informed that their annotations would be used for research purposes. No personal identifying information about annotators is included in the released artifacts or reported analyses.

All members have graduated from a music university or a specialized music school. In terms of professional experience, each individual has experience in at least one of the following: performance, composition, arrangement, serving as a part-time lecturer at a music university, conducting research at a university, or audio analysis.

A.4 Model Training Details

A.4.1 Transition-Aware Encoder Pretraining Details

We initialize the transition-aware music encoder from MERT (Li et al., 2024) and adapt it with LoRA (Hu et al., 2022). The encoder is trained with two auxiliary temporal objectives: transition probability prediction and mood-change prediction.

For transition probability prediction, we use structural segment boundaries obtained from the music segmentation model as supervision. Rather than treating each boundary as a hard frame label, we construct a Gaussian target distribution centered at each boundary timestamp. A lightweight 1D convolutional prediction head is attached to the frame-level MERT representations to estimate transition probabilities over time. This objective encourages the encoder to assign high probability to musically salient change points, such as source entries, rhythmic changes, and structural transitions.

For mood-change prediction, we attach a linear regression head to the encoder and train it to predict the change in emotional intensity around each transition boundary. The supervision comes from the human mood-change annotations described in §3.1. This objective encourages the encoder to capture affective temporal dynamics in addition to structural changes.

A.4.2 Timestamped Caption Pretraining Details

In timestamped caption pretraining, the model uses two complementary audio streams. The frozen Qwen2 audio tower provides semantic audio representations, while the transition-aware MUST encoder provides fine-grained acoustic and temporal representations. Because the two encoders produce representations in different embedding spaces, we use a learnable projector to map MUST encoder outputs into the LLM embedding space.

We add sinusoidal time embeddings to the projected MUST tokens based on their absolute timestamps in the original audio. This allows the LLM to distinguish not only the content of each token but also when it occurs in the track. The projected MUST tokens and Qwen tokens are concatenated sequentially, separated by a special <AUDIO_SPLIT> token, and then provided to the LLM together with the text prompt.

To fit long music tracks into a limited token budget, we apply transition-aware dynamic sampling

using the transition probabilities predicted by the encoder. Uniform sampling provides global coverage but can underrepresent short yet musically important events. We therefore allocate more MUST tokens to regions with high transition probability while preserving coverage over the full track. This allows the model to retain fine-grained information around likely transition points without discarding the global musical context.

A.4.3 Supervised QA Fine-Tuning Objective

Our QA fine-tuning data contain heterogeneous answer formats, including multiple-choice letters, free-form descriptions, single timestamps, and temporal intervals. Because these answer types have substantially different sequence lengths, a standard token-averaged language modeling loss can overemphasize tasks with longer answers, such as transition-aware description. We therefore use an answer-only, sample-normalized, task-balanced supervised fine-tuning objective.

For each training example i , we mask all audio tokens, prompt tokens, instruction tokens, and question tokens with the ignore label, and compute the loss only on the answer tokens. Let $\mathcal{A}_i$ denote the set of supervised answer-token positions for example i , and let $\ell_{i,j}$ be the token-level negative log-likelihood at answer position j . We first normalize the loss within each sample:

$$\mathcal{L}_i = \frac{1}{|\mathcal{A}_i|} \sum_{j \in \mathcal{A}_i} \ell_{i,j}.$$

To prevent tasks with more examples in a minibatch from dominating the update, we then average losses within each QA type and assign equal weight to all QA types present in the minibatch. Let $\mathcal{B}_k$ denote the set of examples of QA type k in the minibatch, and let $\mathcal{T}_\mathcal{B}$ denote the set of QA types that appear in the minibatch. The final supervised fine-tuning loss is:

$$\begin{aligned} \mathcal{L}_{\text{SFT}} &= \frac{1}{|\mathcal{T}_\mathcal{B}|} \sum_{k \in \mathcal{T}_\mathcal{B}} \frac{1}{|\mathcal{B}_k|} \sum_{i \in \mathcal{B}_k} \mathcal{L}_i \\ &= \frac{1}{|\mathcal{T}_\mathcal{B}|} \sum_{k \in \mathcal{T}_\mathcal{B}} \frac{1}{|\mathcal{B}_k|} \sum_{i \in \mathcal{B}_k} \frac{1}{|\mathcal{A}_i|} \sum_{j \in \mathcal{A}_i} \ell_{i,j}. \end{aligned}$$

This objective ensures that supervision is applied only to the target answer, that each example contributes equally regardless of answer length, and that each QA type receives balanced weight during training.

A.4.4 GRPO Reward Details

The main goal is to provide a continuous reward signal instead of relying only on hard threshold-based metrics. This allows predictions that are temporally closer to the gold answer to receive higher rewards, even when they do not exactly satisfy the final evaluation metric.

TSG Reward. TSG requires the model to predict a single timestamp. Let $\hat{t}$ be the predicted timestamp, t^* be the gold timestamp, and L be the input audio duration. We convert the absolute temporal error into an exponential decay reward:

$$r_{\text{TSG}} = \exp\left(-\frac{|\hat{t} - t^*|}{15}\right) - 0.5 \cdot \mathbf{1}_{\text{out}} - 1.0 \cdot \mathbf{1}_{\text{fmt}},$$

where $\mathbf{1}_{\text{out}}$ indicates that the predicted timestamp is outside the valid audio range $[0, L]$, and $\mathbf{1}_{\text{fmt}}$ indicates that the response does not follow the required answer format or cannot be parsed as a valid single timestamp.

The scale factor of 15 seconds controls the temporal tolerance of the reward. When the prediction exactly matches the gold timestamp, the reward approaches 1. As the temporal error increases, the reward decreases smoothly. This provides a denser learning signal than threshold-based metrics. For example, predictions with 4 seconds and 30 seconds of error both fail under Hit@3s, but the exponential reward assigns a higher reward to the 4-second error prediction.

MTR Reward. MTR requires the model to predict mood-related temporal intervals in the form [START-END]. Let P denote the predicted interval set and G denote the gold interval set. We apply Gaussian smoothing with $\sigma = 15$ seconds and radius 60 seconds to obtain softened masks P_{soft} and G_{soft} .

The Gaussian-smoothed soft-F1 score is defined as

$$\text{SoftF1}_{\text{gaussian}}(P, G) = \frac{2\langle P_{\text{soft}}, G_{\text{soft}} \rangle}{\|P_{\text{soft}}\|_2^2 + \|G_{\text{soft}}\|_2^2 + \epsilon},$$

where ϵ is a small constant for numerical stability. The MTR reward is then computed as

$$r_{\text{MTR}} = \text{SoftF1}_{\text{gaussian}}(P, G) - 0.5 \cdot \mathbf{1}_{\text{out}} - 1.0 \cdot \mathbf{1}_{\text{fmt}},$$

where $\mathbf{1}_{\text{out}}$ indicates that any predicted interval lies outside the valid audio range, and $\mathbf{1}_{\text{fmt}}$ indicates that the response does not follow the required interval-list format or cannot be parsed into valid intervals.

We use Gaussian-smoothed soft-F1 because MTR requires predicting both the temporal region

and its boundaries. Hard temporal IoU or F1 can assign nearly zero reward even when the prediction is close to the gold interval but slightly misaligned. In contrast, Gaussian soft-F1 gives partial credit to near-miss predictions, encouraging the model to first locate the correct temporal region and then refine the interval boundaries.

A.5 Additional Ablations and Experimental Details

A.5.1 Additional Ablations

Effect of Zero-shot CoT Prompting. We additionally examine whether zero-shot chain-of-thought prompting improves temporal grounding performance. As shown in Table 6, zero-shot CoT slightly improves the total average score (41.9 to 42.3), but its effect is mixed across metrics.

Effect of LoRA Rank. We further analyze the sensitivity of our model to the LoRA rank used during QA fine-tuning. As shown in Table 7, varying the LoRA rank from 32 to 128 leads to only minor differences in overall performance. LoRA 64 achieves the highest average score, so we use LoRA 64 as our default setting, as it provides stable performance across task types while maintaining a moderate adaptation capacity.

A.5.2 Experimental Details

Training Details. We train our model on $8 \times H200$ GPUs using the AdamW optimizer (Loshchilov and Hutter, 2017) with a cosine learning rate scheduler (Loshchilov and Hutter, 2016). We set MUST token ratio as 6.66 tokens/sec as default. Detailed hyperparameters for each LLM training stage are summarized in Table 11.

Prompting Strategy	TSG		LTR	TAD		GTO	MTR		Total
	Onset Hit@3s	Offset Hit@3s	Acc.	METEOR	CLAPScore	Acc.	Temporal IoU	Temporal F1	Avg.
Without CoT	40.0	63.5	61.1	21.1	34.3	65.7	21.4	28.0	41.9
Zero-shot CoT	41.5	58.0	64.4	21.9	33.3	70.0	21.6	28.0	42.3

Table 6: Effect of zero-shot chain-of-thought prompting. We compare the default prompting strategy without CoT against zero-shot CoT prompting. Zero-shot CoT yields a slightly higher average score, while its effect varies across task-specific metrics.

LoRA Rank	TSG		LTR	TAD		GTO	MTR		Total
	Onset Hit@3s	Offset Hit@3s	Acc.	METEOR	CLAPScore	Acc.	Temporal IoU	Temporal F1	Avg.
LoRA 32	33.5	62.5	59.1	20.6	34.0	68.2	21.6	28.0	40.9
LoRA 64	40.0	63.5	61.1	21.1	34.3	65.7	21.4	28.0	41.9
LoRA 128	37.0	64.5	62.1	20.6	33.3	63.6	21.5	28.0	41.3

Table 7: Ablation study on the LoRA rank used during QA fine-tuning.

Category	Instrument	Count	Rate	Total
Drums	Cymbals	70,074	46.93%	51.68%
	Mallet percussion	5,231	3.50%	
	Drums	1,863	1.25%	
Synthesizer	Synthesizer	18,608	12.46%	12.46%
Plucked strings	Guitar	15,877	10.63%	11.36%
	Banjo	586	0.39%	
	Mandolin	383	0.26%	
	Ukulele	116	0.08%	
Bowed strings	Violin	5,685	3.81%	6.87%
	Cello	4,571	3.06%	
Vocals	Vocals	8,947	5.99%	5.99%
Keys	Piano	7,235	4.84%	5.59%
	Organ	824	0.55%	
	Accordion	283	0.19%	
Winds	Flute	3,155	2.11%	3.29%
	Saxophone	1,735	1.16%	
	Clarinet	17	0.01%	
Brass	Trumpet	1,844	1.23%	1.98%
	Trombone	1,106	0.74%	
Bass	Bass	1,190	0.80%	0.80%
Total	–	149,330	–	100.00%

Table 8: Distribution of added instrument labels grouped into nine instrument categories after cross-validation.

Category	Instrument	Count	Rate	Total
Keys	Piano	30,082	32.03%	32.51%
	Organ	410	0.44%	
	Accordion	43	0.05%	
Bass	Bass	23,871	25.41%	25.41%
Drums	Drums	22,103	23.53%	24.37%
	Mallet percussion	663	0.71%	
	Cymbals	122	0.13%	
Vocals	Voice/vocals	5,848	6.23%	6.23%
Plucked strings	Guitar	4,550	4.84%	5.43%
	Mandolin	473	0.50%	
	Ukulele	57	0.06%	
	Banjo	23	0.02%	
Synthesizer	Synthesizer	4,912	5.23%	5.23%
Bowed strings	Cello	500	0.53%	0.66%
	Violin	116	0.12%	
Winds	Saxophone	96	0.10%	0.14%
	Flute	20	0.02%	
	Clarinet	12	0.01%	
Brass	Trumpet	27	0.03%	0.03%
	Trombone	0	0.00%	
Total	–	93,928	–	100.00%

Table 9: Distribution of removed instrument labels grouped into nine instrument categories after cross-validation.

[000-032] <Description> This music unfolds at a steady 120 BPM in the bright and open key of G major. The overall soundscape is warm and inviting, with a moderate-to-loud presence that creates a sense of gentle momentum. The texture is dominated by a single, continuous melodic instrument that drives the piece forward with a soft, lyrical attack. Its timbre is rich and mellow, suggesting a plucked string instrument like a kora or a harp, playing flowing, arpeggiated patterns. . .

[032] <Boundary> In perceived mood, emotional arousal decreases noticeably. Musically, The overall loudness remains moderate, with a continuous texture before and after. The harmony modulates from a strong G major center to a strong C major center. Rhythmically, the tempo remains steady. Rhythmically, the rhythmic flow stays steady and moderate. Instrument-wise, Other Instrument stays present with a stable role.

[032-154] <Description> This is a vibrant and rhythmically engaging piece in C major with a tempo of 120 BPM. The overall soundscape is warm and inviting, driven by a continuous and steady melodic core. The music is built around a prominent, bright-sounding plucked string instrument—likely a kora or a similar West African harp . . .

[154] <Boundary> In perceived mood, emotional arousal increases slightly. Musically, The overall loudness remains moderate, with a continuous texture before and after. The harmony stays stable around a strong C major center. Rhythmically, the tempo remains steady. Rhythmically, the rhythmic flow stays steady and moderate. Instrument-wise, Vocals enters and becomes audible; Other Instrument stays present with a stable role.

[154-175] <Description> This music segment unfolds at a steady 120 BPM in the bright and open key of C major. The overall soundscape is warm and intimate, with a moderate volume that creates a relaxed yet engaging atmosphere. The texture is driven by a continuous melodic line that forms the core of the piece. A female voice enters with a soft, . . .

[175] <Boundary> In perceived mood, emotional arousal decreases noticeably. Musically, The overall loudness remains moderate, with a continuous texture before and after. The harmony stays stable around a strong C major center. Rhythmically, the rhythmic flow stays steady and moderate, and the tempo undergoes a dramatic shift. Instrument-wise, Vocals drops out of the mix; Other Instrument stays present with a stable role.

[175-245] <Description> This music unfolds at a brisk 150 BPM in the key of C major, creating a vibrant and energetic atmosphere. The overall soundscape is defined by a continuous, driving pulse, with a melodic core . . .

Figure 6: Example of timestamped music caption with static and dynamic captions. (Before rewriting)

[000-032] <Description> This music unfolds at a steady 120 BPM in the bright and open key of G major. The overall soundscape is warm and inviting, with a moderate-to-loud presence that creates a sense of gentle momentum. The texture is dominated by a single, continuous melodic instrument that drives the piece forward with a soft, lyrical attack. Its timbre is rich and mellow, suggesting a plucked string instrument like a kora or a harp, playing flowing, arpeggiated patterns. . .

[032] <Boundary> In perceived mood, emotional arousal decreases noticeably. Musically, The overall loudness remains moderate, with a continuous texture before and after. The harmony modulates from a strong G major center to a strong C major center. Rhythmically, the tempo remains steady. Rhythmically, the rhythmic flow stays steady and moderate. Instrument-wise, Other Instrument stays present with a stable role.

[032-154] <Description> Continuing the warm momentum, the music now settles into a bright C-major center at a steady 120 BPM while the overall arousal gently eases. The continuous, steady melodic core persists, driven by the same luminous plucked-string voice—its clear, resonant timbre shimmering like a West African kora, . . .

[154] <Boundary> In perceived mood, emotional arousal increases slightly. Musically, The overall loudness remains moderate, with a continuous texture before and after. The harmony stays stable around a strong C major center. Rhythmically, the tempo remains steady. Rhythmically, the rhythmic flow stays steady and moderate. Instrument-wise, Vocals enters and becomes audible; Other Instrument stays present with a stable role.

[154-175] <Description> With the slight rise in arousal, a soft, ethereal female voice slips into the texture, its wordless, breathy phrases hovering above the established melodic current. The bright C-major backdrop and steady 120 BPM pulse remain unchanged, and the warm, resonant plucked-string voice that has been threading the piece continues its intricate, finger-plucked arpeggios. . .

[175] <Boundary> In perceived mood, emotional arousal decreases noticeably. Musically, The overall loudness remains moderate, with a continuous texture before and after. The harmony stays stable around a strong C major center. Rhythmically, the rhythmic flow stays steady and moderate, and the tempo undergoes a dramatic shift. Instrument-wise, Vocals drops out of the mix; Other Instrument stays present with a stable role.

[175-245] <Description> As the breathy female vocal fades away, the tempo leaps dramatically to a brisk 150 BPM while the harmony stays rooted in C major. The surge in pulse is tempered by a noticeable drop in emotional arousal, so the music feels calmer even as it propels forward. A continuous, driving pulse maintains the moderate volume and steady texture established earlier. The primary voice remains the plucked-string instrument . . .

Figure 7: Example of timestamped music caption with static and dynamic captions. (After rewriting)

Category	Feature	Description & Usage in Pipeline
Tonal & Harmonic	Key & Scale	Root note and mode (e.g., C Major); used to identify tonal centers and track modulations across boundaries.
	Key Strength	Confidence score (0.0 ~ 1.0) of the key prediction; determines the clarity and ambiguity of the harmony.
	Chroma	12-dimensional pitch class profile; cosine distance between segments measures the degree of harmonic palette shifts.
	Dissonance	Measure of sensory dissonance; utilized to evaluate whether musical tension grows or resolves during a transition.
Rhythmic	BPM	Beats per minute; computes absolute changes (ΔBPM) to detect drastic or subtle tempo accelerations/decelerations.
	Onset Strength Mean	Average intensity of rhythmic attacks; captures the driving force, busyness, and energy of the rhythmic texture.
Energy & Dynamics	Volume	Loudness level used to compute energy shifts and strictly detect silence or instrument entries/exits.
	Active Ratio	The proportion of active audio frames; describes the density and textural busyness of the mix and individual stems.
Timbral & Semantic	Spectral Centroid	The center of mass of the spectrum (in Hz); serves as a proxy for identifying the brightness or darkness of the timbre.

Table 10: Summary of extracted musical features utilized for timestamped music caption generation.

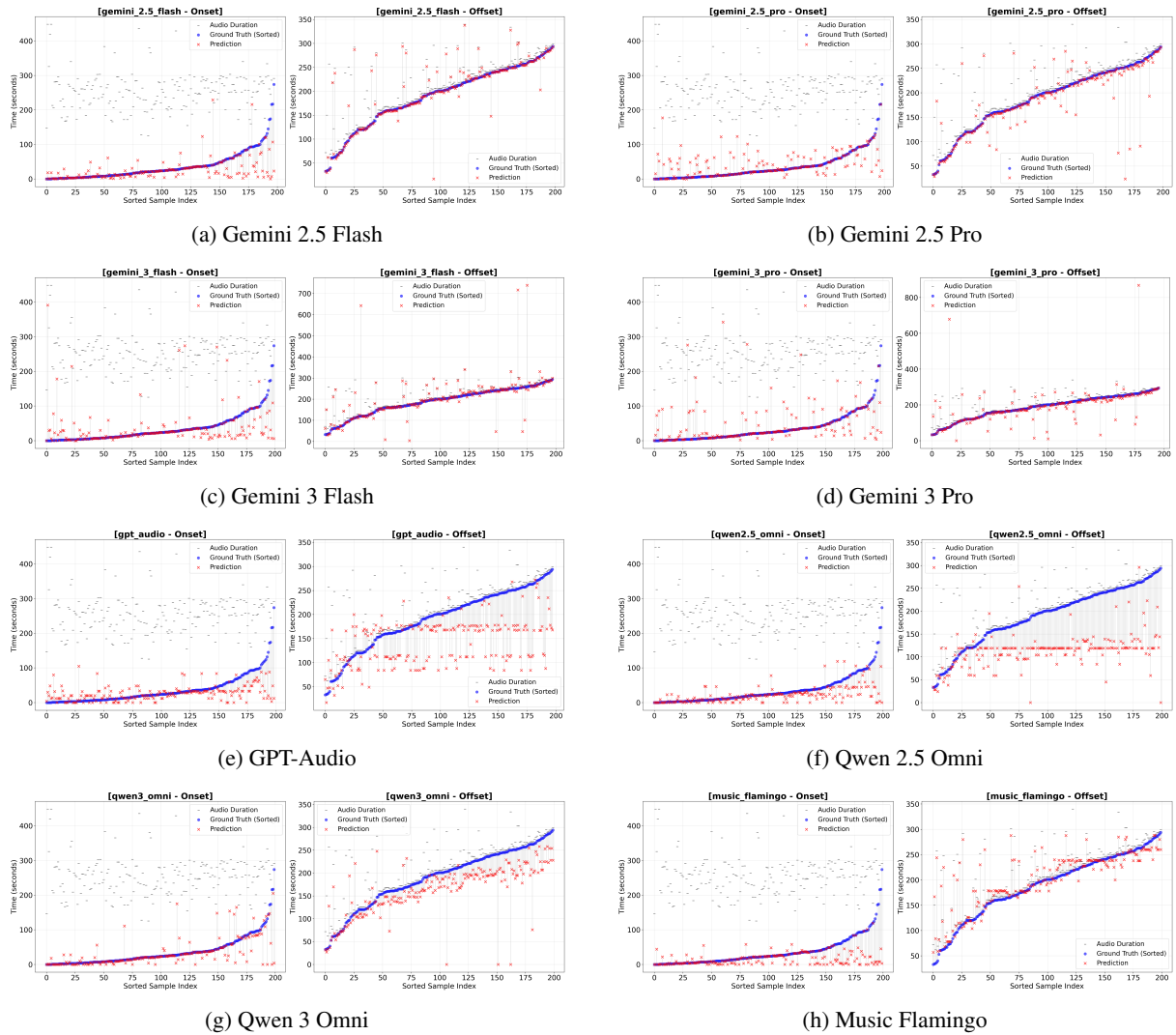


Figure 8: Temporal source grounding (TSG) results across baseline models. ● denotes the ground-truth timestamp, and × denotes the predicted timestamp.

What best describes the musical change at 133s?

- A. . . .
- B. . . .
- C. . . .
- D. . . .

You need to listen around T-10 - T+10, then fill 'answer' column with

(If there is correct answer) correct option → A/B/C/D

(If there's no correct answer) your own correct sentence (description) → There's vocal entry and . . .

Figure 9: Human-assisted validation instruction for LTR and TAD.

The following descriptions (X, Y, Z) represent three distinct musical transitions from the same track.
 Descriptions:
 (X) [200s] ...
 (Y) [15s] ...
 (Z) [160s] ...
 Question: Which of the following options represents the correct chronological order of these transitions as they appear in the track?

- A. X → Y → Z
- B. X → Z → Y
- C. Y → X → Z
- D. Y → Z → X « already known
- E. Z → X → Y
- F. Z → Y → X

You need to listen X, Y, and Z (around T-10 - T+10) to check whether the descriptions are correct.
 If they are correct, please write "Correct".
 If there is a wrong description, please type wrong transition with reason. (ex. X: There is no guitar entry)
 If there are multiple wrong descriptions, please use comma separated values.
 (ex. X: There is no guitar entry, Y: There is no key change)

Figure 10: Human-assisted validation instruction for GTO.

Question
 When is(are) the highest arousal duration(s) in this audio?
 Please answer with duration in seconds (ex. [72-110])
 If there are multiple timestamps for highest/lowest duration, you can list multiple durations.
 Ex) [55-110], [130-160]
 Arousal is the level of energy and intensity.

Figure 11: Human-assisted validation instruction for MTR.

Hyperparameter	Caption Pretraining	QA FT	GRPO
LoRA rank	–	64	64
LoRA alpha	–	128	128
Learning rate	5.0×10^{-6}	5.0×10^{-5}	1.0×10^{-5}
MERT projector learning rate	5.0×10^{-5}	1.0×10^{-5}	1.0×10^{-6}
Effective batch size	16	16	32
Training data	41K	40K	1.6K
Training duration	1 epoch	1 epoch	250 steps

Table 11: Training hyperparameters for each LLM training stage.