MUUNE: TOWARDS FINE-GRAINED FULL-SONG MUSIC RETRIEVAL WITH MULTIMODAL LARGE LANGUAGE MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Music retrieval requires correspondence between complete songs and diverse linguistic expressions, together with recognition of multiple levels of musical attributes, including style, instrumentation, vocals and music theory. Existing methods make substantial progress in short-clip alignment, but preserving full-song detail, understanding complex queries and constructing high-quality supervision remain key challenges. We present **MUUNE** (**MU**sic **UN**derstanding **E**mbedding), which combines music–text data construction, music knowledge adaptation and unified representation learning with a multimodal large language model (MLLM). Audio language models and music information retrieval tools provide structured metadata for multi-view retrieval supervision and atomic counterfactual evaluation. MUUNE adapts a pretrained audio language model into a Music Reasoner and learns shared music–text representations supporting inputs of up to 300 seconds. MUUNE-EH with Music-Meta-of-Thought (MMOT) exceeds all compared baselines on all seven primary retrieval tasks, reaching 44.48% mean R@10 and 93.93% mean counterfactual accuracy, exceeding CLaMP3 by 23.30 and 15.37 percentage points, respectively. Relative to MuQ-MuLan adapted with the same contrastive data, the gains are 16.09 and 7.70 percentage points. Controlled ablations show that music knowledge adaptation and an independent embedding head improve retrieval. With the model and queries fixed, extending audio from 30 to 300 seconds improves the retrieval mean by 10.95 points. The benefit of explicit music analysis depends on the representation design.

1 INTRODUCTION

Music retrieval maps natural language and music into a shared space for search, recommendation and creative reference. Queries range from ‘gentle female vocals, acoustic guitar and melancholic pop’ to ‘piano-led instrumental music in C minor, around 110 BPM, in 4/4 time’. They combine global semantics with specialist constraints on key, tempo and metre (Anand et al., 2026), expressed as keywords, natural queries or detailed descriptions. Text-to-music generation further motivates retrieving reference works from precise descriptions (Agostinelli et al., 2023). Retrieval must therefore match both song-level semantics and specific musical conditions across the full recording.

Contrastive audio–language models establish shared music–text spaces (Wu et al., 2023; Huang et al., 2022; Zhu et al., 2025; Wu et al., 2025b), but common CLAP- and MuLan-style models aggregate approximately 10-second windows for full-song retrieval, limiting explicit modelling of cross-window relationships. Global alignment also struggles to exploit multiple conditions in long descriptions, motivating finer alignment (Anand et al., 2026). Generative models offer a complementary route: their semantic and reasoning capabilities support retrieval representations in language and vision (Tang et al., 2026; Lan et al., 2026), audio–text retrieval (Xu et al., 2026; Yoo et al., 2026) and context-enhanced multimodal embeddings (Cui et al., 2026). This motivates a unified MLLM for long audio and complex music queries.

High-quality music–text data present a second bottleneck: copyright and distribution conditions constrain open resources, while accurate musical annotation requires expertise. Existing resources use human descriptions, web text and tag-to-description conversion (Agostinelli et al., 2023; Manco

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

et al., 2023; Weck et al., 2024; Doh et al., 2023; Roy et al., 2025). Fine-grained full-song retrieval additionally requires rich attributes, consistency across expressions and scalable production. Audio language models and metadata-based pipelines make these requirements more attainable by combining semantic analysis, MIR and language generation (Anand et al., 2026; Bukey et al., 2026; Xu et al., 2025; OpenMOSS Team, 2026a; wx9songs, 2026), motivating a shared attribute layer for diverse retrieval supervision.

To meet these requirements, we introduce **MUUNE** (**MU**sic **UN**derstanding **E**mbedding). The framework turns shared musical attributes into varied retrieval texts, adapts an audio–language model to this domain knowledge, and learns a common embedding space for fine-grained full-song search.

Our contributions are as follows: **1)** We introduce a unified music–text representation framework supporting audio of up to 300 seconds, and compare EOS-LoRA and an independent embedding head (EH) with and without explicit music analysis; **2)** We construct a scalable multi-view supervision pipeline and atomic counterfactual evaluations, with public-data extensions based on SDD, MTG-Jamendo, Music4All and JamendoMaxCaps. The supervision benefits multiple representation models; we plan to release the evaluation resources and construction protocols; **3.** MUUNE-EH with MMOT exceeds the public and data-adapted baselines across seven primary retrieval tasks. Ablations establish the contributions of music knowledge and representation capacity, while showing that explicit analysis depends on the representation scheme.

2 Related Work

Audio–language and music retrieval representations. Contrastive learning enables open-text retrieval: CLAP combines feature fusion and text augmentation (Wu et al., 2023); MuLan aligns music with free-form text using dual encoders (Huang et al., 2022); MuQ-MuLan builds on MuQ's self-supervised acoustics (Zhu et al., 2025); and CLaMP3 extends modality, language and duration coverage (Wu et al., 2025b). HeartCLAP adds joint tag/text training and attribute masking to MuQ-MuLan (HeartMuLa Team, 2026). FIGMA improves long-description attribute matching through global, frame- and token-level objectives (Anand et al., 2026). For long audio, CLAP/MuLan-style models average short windows, while CoLLAP uses segment encoding and attention-based temporal fusion (Wu et al., 2025a). MUUNE encodes the song within one music-adapted language-model sequence.

From generative multimodal models to retrieval representations. Generative backbones transfer semantic, contextual and instruction understanding to retrieval. AuroLA combines multi-granularity supervision, special tokens, Hybrid-NCE and reranking (Xu et al., 2026); Omni-Embed-Audio uses LoRA, mean pooling and modality projections for questions, commands, keywords and exclusion queries (Yoo et al., 2026); WAVE unifies audio/video/text through cross-layer final-token fusion (Tang et al., 2026). Generated context enhances Think Then Embed (Cui et al., 2026); UME-R1 studies generative/discriminative complementarity through supervised fine-tuning (SFT) and reinforcement learning (Lan et al., 2026); LaME combines reason-token bottlenecks with reconstruction/retrieval heads (Wu et al., 2026). MUUNE compares two representation schemes on the same music-adapted backbone, using structured music analysis as context.

Music understanding and music knowledge adaptation. Within the Qwen3-Omni audio-understanding family, Qwen3-ASR provides our initialisation (Xu et al., 2025; Shi et al., 2026). Music Flamingo combines full-song audio, MIR and multi-stage training for instrumentation, lyrics, harmony and complex question answering (Ghosh et al., 2026); TinyMU targets compact music understanding (Li et al., 2026); GaMMA jointly models global/temporal information (You et al., 2026); MOSS-Music supports captioning, lyrics and music-theoretic analysis (OpenMOSS Team, 2026a). MUUNE transfers metadata-supervised domain adaptation into retrieval representations.

Music–text resources and automatic annotation. MusicCaps and SDD provide human descriptions for language-generalisation evaluation (Agostinelli et al., 2023; Manco et al., 2023); WikiMuTe filters short and long Wikipedia music text (Weck et al., 2024); LP-MusicCaps uses an LLM to convert tags into captions (Doh et al., 2023); JamendoMaxCaps scales captioning with metadata imputation (Roy et al., 2025). MusicBench adds tempo, metre, key and chords (Melechovsky et al., 2024); FGMCaps combines MIR and LLMs for fine-grained captions (Anand et al., 2026). Rethink-

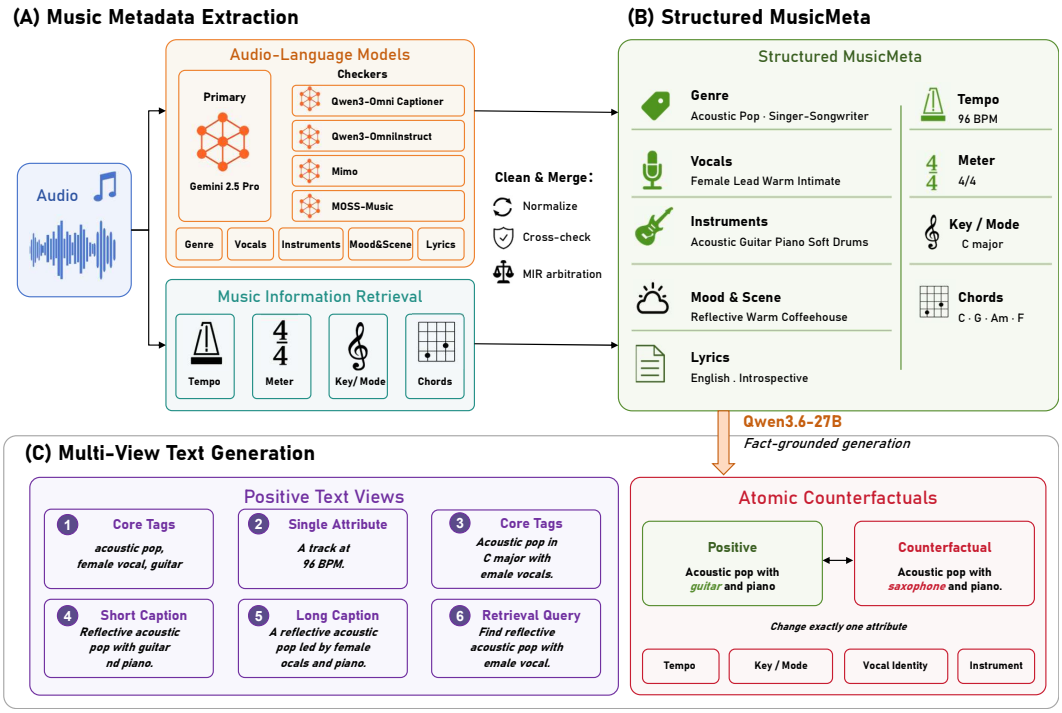


Figure 1: Construction of multi-view fine-grained full-song data. Audio language models and MIR extract semantic and music-theoretic attributes, respectively. Cleaning and consolidation produce MusicMeta, from which the pipeline generates multi-view positive texts and atomic counterfactuals.

ing Music Captioning separates metadata prediction from expression (Bukey et al., 2026), while SonicVerse incorporates attribute prediction (Chopra et al., 2025). TTMR++ enriches text with music metadata, entities and relative similarity (Doh et al., 2024). MOSS-Audio describes event-preserving, branch-specific annotation (OpenMOSS Team, 2026b); MOSS-Music-Data-Pipeline documents music-model, MIR and text-generation integration (wx9songs, 2026). MUUNE adds a shared fact layer, cross-source attribute checks, specialist conditions and music-domain SFT, selecting one fixed positive per song and reserving atomic counterfactuals for evaluation.

3 Multi-View Fine-Grained Full-Song Data

3.1 Multi-Attribute Music Metadata

The pipeline separates audio-grounded attribute extraction from textual expression, providing shared musical facts for both knowledge adaptation and retrieval supervision.

The semantic branch uses Gemini 2.5 Pro (Gemini Team, 2025) as its primary annotation source and models including Qwen3-Omni (Xu et al., 2025), MiMo-Audio (Xiaomi MiMo Team, 2025) and MOSS-Music (OpenMOSS Team, 2026a) for cross-verification. It extracts genre, vocals, instrumentation, mood and scene, and lyrics. Consolidation anchors on the primary annotation, uses cross-source agreement to correct errors and fill gaps, and normalises synonymous tags and field structures.

The MIR branch uses BeatNet (Heydari et al., 2021), Chordino (Mauch & Dixon, 2010) and Essentia (Bogdanov et al., 2013) to estimate BPM, metre, key and chords, following the integrated pipeline (wx9songs, 2026). Semantic and MIR outputs align by audio ID and undergo normalisation and conflict resolution to form MusicMeta. Its fields cover genre, vocals, instrumentation, rhythm, harmony, mood and lyrics; for example, a record may describe acoustic pop with female vocals, guitar and piano, 96 BPM and C major. The same facts supervise the Reasoner and condition retrieval-text generation.

MusicMeta groups genre and style, vocal identity/timbre/delivery, dominant instruments and arrangement roles, BPM and time signature, key/mode/chords, mood and listening context, and lyric

language/content. These fields support category, vocal, timbral, rhythmic, harmonic and intent-based retrieval; lyrics provide music-description supervision for the Reasoner.

This design combines the semantic coverage of audio language models with MIR attribute analysis, providing a common basis for subsequent retrieval texts. Appendices A.2 and A.4 detail the model sources, consolidation prompts and field constraints.

3.2 Multi-View Retrieval Texts

Qwen3.6-27B (Qwen Team, 2026) converts each MusicMeta record into ten English views: tags, single and combined attributes, short and long captions, and natural and specialist queries. For example, shared facts can yield 'Acoustic pop; female vocal; guitar' or 'Find reflective acoustic pop with female vocals.' Table 4 defines the views and seven strong-positive candidate types.

Texts preserve normalised musical facts, record their source fields and describe song-level properties. Song titles, artists and verbatim lyrics are excluded as retrieval cues.

After structural and factual checks, evaluation-set exclusion and global text deduplication, we select one strong-positive text per song to form one-to-one training pairs. This selection remains fixed, while other views support diagnosis and evaluation. Human audits give query hallucination rates of 2.80% for SDD-MV and 2.55% for MTG-Test-MV (Appendix A.5).

3.3 Atomic Counterfactuals and Public Evaluation Extensions

To test whether vectors recognise key query conditions, we construct four types of atomic counterfactuals: tempo, key or mode, vocal identity, and dominant instrument. Each positive–negative text pair changes only one attribute while preserving the remaining content and sentence structure. Evaluation thus directly tests preference for audio-consistent attributes without relying on overall descriptive style. Counterfactuals serve evaluation; standard training negatives come from other songs within the batch.

We further construct SDD-MV and MTG-Test-MV multi-view and counterfactual evaluations from public audio, produce multiple retrieval texts for Music4All (Pegoraro Santana et al., 2020), and extend JamendoMaxCaps (Roy et al., 2025) with full-song descriptions. These resources bring natural expressions, full-song semantics and controllable attribute conditions into one evaluation framework. Appendix A.3 provides examples and protocols. We plan to release the evaluation resources and construction protocols.

4 Music-Adapted Representation Learning

4.1 Music Reasoner and MMOT

MUUNE first adapts a pretrained audio language model into a Music Reasoner through music metadata supervision. Given audio and a fixed analysis instruction, the model learns to predict structured information including genre, vocals, instrumentation, tempo, key, mood and lyrics. This training connects acoustic representations with musical concepts, providing domain knowledge for retrieval.

We call the Reasoner's explicitly generated structured analysis *Music-Meta-of-Thought* (MMOT). Drawing on intermediate semantic expressions in Chain-of-Thought, MMOT organises the predicted musical attributes into metadata. Section 4.3 describes how this analysis conditions retrieval.

4.2 Shared Audio–Text Representations

An encoder converts audio into token sequences that share the MLLM's language-modelling pathway with text. The model aggregates longer-range musical information within a sequence and maps audio and queries to normalised vectors in a shared space. Let x denote the policy-specific input sequence and R_θ the Music Reasoner. Because both sides use shared representation parameters, their outputs are directly comparable by dot product. We compare two representation schemes based on the same Music Reasoner.

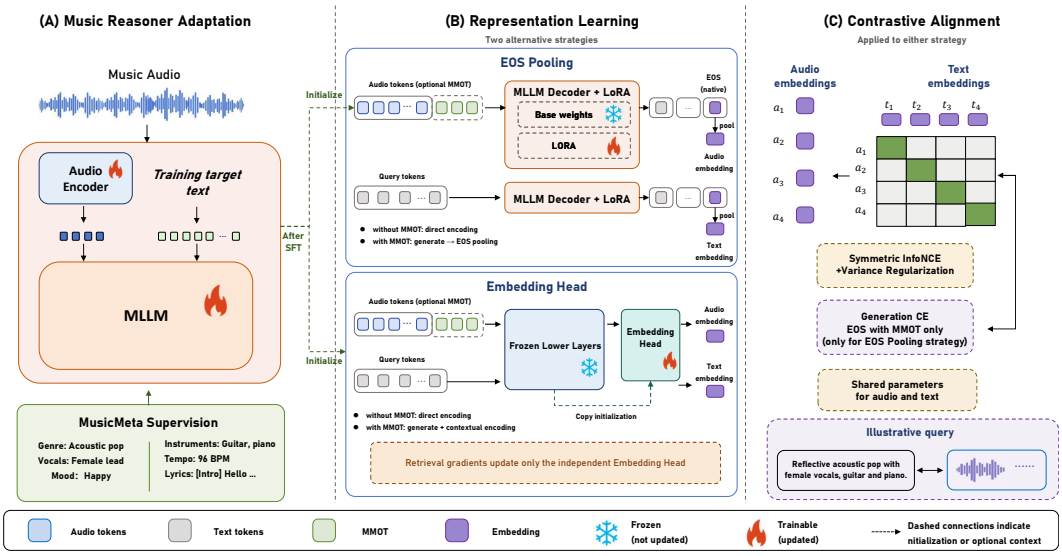


Figure 2: MUUNE training and representation framework. Music metadata supervision first adapts the Music Reasoner; EOS-LoRA or an independent EH then learns unified music–text vectors. MMOT provides optional explicit music analysis conditioning.

EOS-LoRA. We freeze the original model weights, add a low-rank update ΔW to the attention and feed-forward projections of the language decoder, and read the last-layer hidden state of the final valid EOS token:

$$\Delta W = \frac{\alpha}{r} BA, \qquad z^{\mathrm{EOS}}(x) = \mathrm{norm}_2(R_{\theta,\Delta\theta}(x)_{p_{\mathrm{EOS}}}), \tag{1}$$

where p_{EOS} is the position of the EOS token in the valid sequence. Audio and text share the same LoRA; with MMOT, metadata generation and retrieval representation are jointly optimised through these low-rank parameters.

Independent Embedding Head. We freeze the Music Reasoner and initialise a trainable embedding head H_ϕ from its final K Transformer layers and final normalisation layer. Letting L denote the decoder depth,

$$z^{\mathrm{EH}}(x) = \mathrm{norm}_2\Big[H_\phi\big(\mathrm{stopgrad}\big(R_\theta^{1:L-K}(x)\big)\big)_{p_{\mathrm{EH}}}\Big]. \tag{2}$$

The representation forward pass first traverses the frozen earlier $L - K$ layers, after which the EH aggregates retrieval-relevant information; stopgrad indicates that retrieval gradients do not update the Reasoner. Audio and text share one EH with no modality-specific projection, so the Reasoner retains its analysis capability while the retrieval branch obtains an independent optimisation space.

Both schemes contain approximately 403M trainable parameters, enabling comparison at similar adaptation capacities. The principal EH configuration contains eight layers and outputs 2048-dimensional vectors. Appendix B provides model specifications, pooling positions and parameter settings.

4.3 Using Explicit Music Analysis

Without MMOT: audio and the retrieval instruction undergo direct encoding, and the model forms representations using musical knowledge in its parameters.

With MMOT: EOS-LoRA generates MMOT and then reads the final EOS representation, jointly optimising generation and retrieval during training. For EH, the frozen Reasoner first generates MMOT; the original audio, music analysis and retrieval instruction then undergo joint encoding, with retrieval training updating only the EH. Text queries undergo direct encoding in both configurations.

These input settings allow us to examine the additional role of explicit analysis beyond music knowledge adaptation. MMOT can be generated and cached before index construction. Appendix B.2–B.3 provides message structures and context handling details.

4.4 Training Objectives

Both schemes use bidirectional audio–text contrastive learning to bring paired vectors closer and use other songs as negatives. Training handles multiple positives by song ID and adds variance regularisation to maintain useful variation across representation dimensions. The overall objective is:

$$\mathcal{L}_{\text{rep}} = \mathcal{L}_{\text{CL}} + \lambda_{\text{var}}\mathcal{L}_{\text{var}} + \mathbf{1}_{\text{EOS+MMOT}}\,\lambda_{\text{CE}}\mathcal{L}_{\text{CE}}. \tag{3}$$

Generative cross-entropy applies only to EOS-LoRA with MMOT; the EH stage trains the independent representation branch using contrastive and variance objectives. Appendix B.4 defines the losses and their weights.

5 Experimental Setup

Comparison models and training settings. Public baselines include LAION-CLAP with music weights (Wu et al., 2023), MuQ-MuLan, CLaMP3, FIGMA, WAVE-7B (Tang et al., 2026) and AuroLA-7B (Xu et al., 2026). The latter two provide direct comparisons with MLLM-based retrieval representations. We assess data transfer by adapting CLAP and MuQ-MuLan with identical CL-Subset/CL-Full contrastive data, and compare EOS-LoRA and EH, each with or without MMOT. Music knowledge and data-scaling ablations separately examine music SFT and contrastive training scales.

All models use identical candidate sets, queries and scoring rules within each task, while retaining their respective audio encoding methods. CLAP, MuQ-MuLan and CLaMP3 use full-song representation or aggregation protocols; MUUNE uses long-sequence encoding of up to 300 seconds.

SFT and CL denote metadata adaptation and contrastive alignment, respectively; Subset has 500K songs and Full has 2.3M songs (approximately 140K hours). Our training data comprise commercial songs from our in-house music-streaming platform; to our knowledge, their distribution differs from that of academic datasets such as SDD and MTG-Jamendo.

Retrieval and counterfactual evaluation. Seven primary retrieval tasks cover human natural-language descriptions, full-song descriptions, multi-view conditions and in-domain search. Table 6 in Appendix A.3 reports task definitions and candidate counts; Appendix A.3.5 gives audio-duration distributions and coverage under the 300-second budget.

For queries with multiple positives, we use the best positive similarity s^+ to compute the strict rank $\text{rank} = 1 + \#\{j : s_j > s^+\}$. Bi-R@10 is the equally weighted mean of text-to-audio and audio-to-text retrieval, and Macro averages equally across the specified views. The seven-task mean weights the seven primary task metrics equally.

Counterfactual evaluation uses 2,488 positive–negative text pairs from SDD-MV and 3,327 from MTG-Test-MV. A prediction counts as correct only if the audio–positive similarity strictly exceeds the audio–negative similarity:

$$\text{CFAcc} = \frac{1}{N}\sum_i \mathbf{1}\left[s(a_i, t_i^+) > s(a_i, t_i^-)\right]. \tag{4}$$

The CF two-task mean averages strict accuracy equally across the two tasks, with ties counted as errors. Retrieval and counterfactual evaluation measure candidate localisation and key-attribute discrimination, respectively, providing complementary assessments.

Supplementary evaluation. We evaluate short-clip retrieval on MusicCaps-test, FGMCaps and MusicBench A/B, and assess classification transfer using frozen-encoder MLP probes on six MARBLE tasks (Yuan et al., 2023), obtaining competitive results. The appendix also reports EH-depth and trainable-capacity ablations. Appendices A and C provide data examples, probe configurations and supplementary results.

Table 1: Retrieval, data scaling and counterfactual results (%). Shaded columns are task means; bold marks the best scores. Groups are public models, data-adapted dual encoders, MUUNE / CL-Subset and MUUNE / CL-Full. MUUNE uses SFT-Full; its headline CF scores use CL-Full, and the CL-Subset EH row additionally reports CF. EOS denotes EOS-LoRA; w/o and w/ denote without and with MMOT. — indicates unreported scores.

Model / setting	Overall		Retrieval by task							Counterfactuals	
	Retr. mean	CF mean	SDD orig.	JMC 30s	JMC fullsong	Music4All US10K	In-house	SDD-MV	MTG-Test MV	SDD-MV CF	MTG-Test MV CF
CLAP	13.66	63.80	39.28	2.98	4.05	2.29	3.15	25.82	18.05	66.56	61.05
MuQ-MuLan	18.04	67.49	50.00	5.34	7.85	3.98	4.81	30.11	24.16	66.84	68.14
CLaMP3	21.18	78.56	44.10	4.60	6.72	9.17	12.73	39.45	31.52	79.86	77.25
FIGMA	8.78	70.85	17.96	3.66	1.55	4.55	2.03	18.80	12.88	71.02	70.69
WAVE-7B	19.45	65.09	49.06	4.26	14.41	4.99	7.98	31.06	24.39	65.31	64.86
AuroLA-7B	22.25	67.72	49.60	6.98	23.57	7.23	7.50	34.47	26.37	68.81	66.64
CLAP / CL-Subset	19.79	84.51	24.66	5.83	3.42	9.38	17.20	44.13	33.94	86.09	82.93
CLAP / CL-Full	21.66	85.01	26.27	6.71	4.17	10.46	19.70	47.70	36.60	86.70	83.32
MuQ-MuLan / CL-Subset	24.72	80.38	45.58	6.17	11.88	9.90	17.50	44.96	37.05	80.75	80.01
MuQ-MuLan / CL-Full	28.39	86.23	44.10	7.56	12.60	12.05	24.48	53.97	43.97	86.33	86.14
EH w/o (CL-Subset)	42.42	91.96	51.74	8.72	24.99	12.91	54.07	76.28	68.25	92.85	91.07
EOS w/o MMOT	39.10	89.69	53.62	7.89	24.07	10.89	45.83	70.41	60.99	90.31	89.06
EOS w/ MMOT	40.97	93.02	**54.42**	7.82	24.27	13.39	49.49	72.92	64.49	93.37	92.67
EH w/o MMOT	44.34	93.58	53.75	**9.19**	**25.61**	15.03	56.49	78.86	71.41	93.77	93.39
EH w/ MMOT	**44.48**	**93.93**	52.95	8.38	24.82	**15.33**	**58.36**	**79.26**	**72.27**	**94.29**	**93.57**

6 Experimental Results

Full-song retrieval and fine-grained attribute discrimination. Table 1 reports the seven retrieval and two atomic counterfactual tasks. MUUNE-EH with MMOT exceeds all compared baselines on all seven retrieval tasks.

MUUNE-EH with MMOT reaches 44.48% mean R@10 and 93.93% counterfactual accuracy, exceeding CLaMP3 by 23.30 and 15.37 percentage points, respectively. WAVE-7B and AuroLA-7B reach retrieval means of 19.45% and 22.25%, and counterfactual means of 65.09% and 67.72%. AuroLA-7B scores 49.60% on SDD_orig and leads the public-weight baselines on JMC-fullsong (23.57%); MUUNE-EH with MMOT exceeds its retrieval mean by 22.23 points. These comparisons establish an advantage over other MLLM retrieval models; the controlled SFT ablation below isolates the contribution of music knowledge adaptation.

Shared MLLM representations further realise the value of the same contrastive data. Relative to MuQ-MuLan / CL-Full, EH with MMOT improves retrieval and counterfactual means by 16.09 and 7.70 percentage points, respectively. Retrieval gains on JMC-fullsong, In-house, SDD-MV and MTG-Test-MV are 12.22, 33.88, 25.29 and 28.30 points. Improvements concentrate on tasks rich in full-song content and multi-attribute conditions, indicating that unified representations with music-specific adaptation make more effective use of this supervision.

EOS with MMOT leads on SDD's human descriptions (54.42%), EH without MMOT leads on both JMC settings, and EH with MMOT leads on Music4All, In-house and both multi-view tasks. The latter also exceeds 93% on each counterfactual task.

Data transfer and scaling effects. We adapt CLAP, MuQ-MuLan and MUUNE-EH with data from the same pipeline and compare public weights, CL-Subset and CL-Full settings. Here, CL-Subset and CL-Full refer to contrastive representation training data. MUUNE-EH keeps its SFT-Full backbone and without-MMOT setting fixed to isolate the effect of contrastive data scale. Table 1 reports both the per-task results and seven-task means.

Pipeline data transfer across architectures: adapting public CLAP and MuQ-MuLan to CL-Full raises their means by 8.00 and 10.35 points. Gains on SDD-MV/MTG-Test-MV are 21.88/18.55 and 23.86/19.81 points, respectively, supporting improved fine-grained condition matching.

Increasing CL-Subset to CL-Full raises the means of CLAP, MuQ-MuLan and MUUNE-EH by 1.87, 3.67 and 1.92 points; MUUNE-EH improves on all seven tasks. With only CL-Subset, MUUNE-EH reaches 42.42%, exceeding MuQ-MuLan / CL-Full (28.39%). Domain adaptation and unified

modelling thus improve supervision use, although transfer to original natural descriptions varies across models.

Public short-clip retrieval. MusicCaps-test contains 2,313 available official test clips; FGMCaps has 10,000 fine-grained pairs; MusicBench A/B each have 400 pairs, with B adding music-theoretic conditions. Each uses a fixed candidate pool and bidirectional R@10 (Table 2).

Table 2: Short-clip bidirectional R@10 (%). MUUNE-EH: SFT-Full, CL-Full, no MMOT.

Model	MusicCaps test	FGMCaps	Music Bench A	Music Bench B
CLAP public	20.97	1.65	62.38	60.88
CLAP / CL-Full	12.60	32.58	42.87	60.62
MuQ public	19.28	6.18	53.63	52.25
MuQ-MuLan / CL-Full	**23.71**	24.56	59.75	74.25
FIGMA	13.90	**63.90**	**78.38**	**83.88**
MUUNE-EH / CL-Full	22.20	29.21	60.63	**83.88**

Musical attribute supervision transfers to short-clip conditions. Data adaptation improves MuQ-MuLan on all four tasks, with FGMCaps increasing from 6.18% to 24.56% and MusicBench B from 52.25% to 74.25%. MUUNE-EH achieves 83.88% on MusicBench B, tying FIGMA for the highest score and demonstrating effective matching of dense music-theoretic conditions. It achieves 22.20% on MusicCaps-test, exceeding FIGMA by 8.30 percentage points.

FIGMA's strength on FGMCaps and MusicBench A complements MUUNE's full-song results, motivating joint local and song-level alignment. CLaMP3 discusses overlap among MusicCaps, SDD and CLAP training data (Wu et al., 2025b); possible overlap and distribution alignment make these public transfer comparisons rather than definitive model rankings.

Model components and ablation analysis. The independent EH outperforms EOS-LoRA at similar adaptation capacity. Without MMOT, EH exceeds EOS-LoRA by 5.24 and 3.89 percentage points in retrieval and counterfactual means; with MMOT, the gains remain 3.51 and 0.91 points. Consistent advantages under both input conditions support a dedicated representation optimisation module for retrieval on a music-adapted backbone.

The benefits of explicit analysis depend on the representation scheme. MMOT increases the retrieval and counterfactual means of EOS by 1.87 and 3.33 percentage points. For EH, retrieval changes only from 44.34% to 44.48% (+0.14 points), and counterfactual accuracy increases by 0.35 points. These small differences show no clear overall gain for EH; task changes are mixed and no significance test is claimed. Direct EH encoding remains attractive for index construction because it avoids explicit generation.

Music knowledge adaptation. Table 3(a) compares initialisations and music SFT using EH and CL-Subset. Qwen3 AuT follows the same audio–language design as the Qwen3-ASR Baseline, pairing a Qwen3-Omni AuT encoder with Qwen3-1.7B. MOSS-Music provides an 8B music-language initialisation (OpenMOSS Team, 2026a). The Baseline/SFT comparison uses the principal eight-layer EH, with 402.69M trainable parameters (Appendix B.1).

Music knowledge adaptation provides sustained, broad retrieval gains. The Baseline seven-task mean is 33.63%, increasing to 37.92% and 42.42% with SFT-Subset and SFT-Full, respectively: gains of 4.29 and 8.79 percentage points. SFT-Full improves all seven metrics over the unadapted model, with JMC-fullsong rising from 8.08% to 24.99%. Structured music supervision therefore effectively strengthens subsequent full-song semantic representations.

Moving from SFT-Subset to SFT-Full further increases the mean by 4.50 percentage points. Together with the multi-view supervision analysis above, these results show complementary roles for the two forms of supervision: music SFT establishes domain knowledge, and contrastive learning organises that knowledge into a retrievable vector space.

Audio input duration. We fix the same trained EH model, query texts, candidate sets and scoring rules, varying only the audio limit from 30 to 60, 120 and 300 seconds (Table 3(b)). Each input starts at its prescribed offset; shorter recordings retain all available audio. All settings use SFT-Full + CL-Full without MMOT and require no retraining.

Table 3: Training and input ablations (%). Panels vary (a) initialisation/SFT, (b) audio-duration limit and (c) second-pass input. The principal EH has eight layers; bold marks panel maxima. *Default setting used in the main results.

Model / setting	SDD orig.	JMC 30s	JMC fullsong	Music4All US10K	In-house	SDD-MV	MTG-Test MV	Mean
(a) Initialisation and music knowledge adaptation; CL-Subset								
Qwen3 AuT	24.93	2.61	4.50	5.34	30.60	52.89	43.68	23.51
MOSS-Music	46.38	5.34	19.75	7.52	39.28	60.18	51.80	32.89
Baseline	41.55	6.63	8.08	8.23	45.93	65.23	59.72	33.63
Baseline / SFT-Subset	46.38	7.60	22.15	10.71	47.61	69.91	61.07	37.92
Baseline / SFT-Full	**51.74**	**8.72**	**24.99**	**12.91**	**54.07**	**76.28**	**68.25**	**42.42**
(b) Audio-duration limit; model and query texts fixed								
30 seconds	45.44	9.19	16.83	15.03	32.48	63.66	51.03	33.38
60 seconds	51.74	9.19	21.62	15.03	45.32	74.65	62.33	39.98
120 seconds	**53.75**	9.19	24.65	15.03	**54.24**	**78.86**	68.46	43.45
300 seconds*	**53.75**	9.19	**25.61**	15.03	56.49	**78.86**	**71.41**	**44.33**
(c) Inference input; fixed EH trained with MMOT								
Audio + matched MMOT	**52.41**	**8.14**	24.30	**14.01**	**53.01**	**75.25**	**68.06**	**42.17**
Audio only	52.01	6.86	**25.29**	9.24	44.37	67.66	59.96	37.91
Audio + shuffled MMOT	48.93	4.83	19.62	6.57	41.22	65.79	56.71	34.81
Matched MMOT only	39.54	5.96	12.67	9.79	41.69	64.90	56.08	32.95
Shuffled MMOT only	11.80	0.07	0.07	0.15	0.09	6.53	0.58	2.76

The retrieval mean rises from 33.38% to 39.98%, 43.45% and 44.33%, a 10.95-point gain from 30 to 300 seconds with unchanged text. Five tasks improve and two (JMC-30s and Music4all-US10K) stay flat, because their source clips are approximately 30 seconds. Gains diminish with duration: 120 seconds is only 0.88 points below 300 seconds. Counterfactual means similarly rise from 89.35% to 92.16%, 93.19% and 93.58%; Appendix C.3 reports individual scores. These results show that additional audio information improves song matching. EH depth results are in Appendix C.2.

Complementarity of audio and MMOT. We fix the SFT-Full Reasoner and the EH trained on CL-Subset with MMOT, varying only the second-pass representation input to compare audio and explicit analysis. All settings retain the same trained weights; the without-MMOT configurations in Table 1 undergo independent training.

The five settings are audio with matched MMOT, audio only, audio with shuffled MMOT, matched MMOT only and shuffled MMOT only. Shuffling is deterministic within each local logical batch, preserves the same MMOT text set and excludes self-pairings, thereby changing the correspondence between music analysis and the current audio.

Table 3(c) reports retrieval, while Figure 3 shows counterfactual results. These differences measure input dependence of a model trained with MMOT; they do not represent gains over an independently trained direct encoder. Complete input reaches 42.17% retrieval and 92.55% counterfactual means. Removing MMOT reduces these by 4.26 and 14.74 points; mismatching it reduces them by 7.36 and 11.77 points. Keeping the text set but breaking its audio correspondence substantially degrades performance, showing reliance on matched musical semantics.

Audio and explicit attributes provide complementary information. Matched MMOT alone retains 88.42% counterfactual accuracy, but its retrieval mean is 32.95%, 9.22 points below complete input. Structured analysis thus preserves substantial attribute information, while audio remains important for distinguishing songs. Shuffled MMOT alone yields only 2.76% retrieval and 45.74% counterfactual means, further demonstrating the importance of correct semantic correspondence.

7 Conclusion and Discussion

MUUNE connects music understanding with fine-grained full-song retrieval through structured supervision, music knowledge adaptation and unified MLLM representations. Comparisons with public and data-adapted baselines, including other MLLM retrieval models, support dedicated representation learning on a music-adapted backbone. Controlled duration ablations show a 10.95-point retrieval gain from 30 to 300 seconds, with diminishing returns. EH without MMOT provides strong retrieval while avoiding generation during index construction; MMOT raises its observed retrieval mean by only 0.14 points, with no clear overall gain. The fixed-model input ablation instead shows dependence on matched analysis when trained with MMOT. Evaluation concerns song-level retrieval within a 300-second budget, whose coverage varies across datasets (Appendix A.3.5), and automatic metadata retain annotation noise. Combining song-level semantics with local alignment, and extending the MARBLE attribute transfer, are directions for future work.

8 AI USE STATEMENT

We use AI tools to polish the manuscript's writing and to assist in generating synthetic music–text datasets for training. We design the data-generation procedures, assess data quality and remain responsible for the experiments, analyses, interpretations and final text.

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. MusicLM: Generating Music From Text. arXiv:2301.11325, 2023. URL https://arxiv.org/abs/2301.11325. Source of the MusicCaps dataset.

Nishit Anand, Ashish Seth, Sreyan Ghosh, Dinesh Manocha, and Ramani Duraiswami. FIGMA: Towards FIne-grained music retrievAl. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 47559–47572. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.2197. URL https://aclanthology.org/2026.acl-long.2197/.

Dmitry Bogdanov, Nicolas Wack, Emilia Gómez, Sankalp Gulati, Perfecto Herrera, Oscar Mayor, Gerard Roma, Justin Salamon, José Zapata, and Xavier Serra. ESSENTIA: An Audio Analysis Library for Music Information Retrieval. In *Proceedings of the 14th International Society for Music Information Retrieval Conference (ISMIR)*, pp. 493–498, 2013. URL https://essentia.upf.edu/.

Dmitry Bogdanov, Minz Won, Philip Tovstogan, Alastair Porter, and Xavier Serra. The MTG-Jamendo Dataset for Automatic Music Tagging. In *ICML Workshop on Machine Learning for Music Discovery*, 2019. URL https://mtg.github.io/mtg-jamendo-dataset/.

Irmak Bukey, Zhepei Wang, Chris Donahue, and Nicholas J. Bryan. Rethinking Music Captioning with Music Metadata LLMS. In *ICASSP 2026 - 2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 15732–15736, 2026. doi: 10.1109/icassp55912.2026.11464125. URL https://doi.org/10.1109/icassp55912.2026.11464125.

Anuradha Chopra, Abhinaba Roy, and Dorien Herremans. SonicVerse: Multi-Task Learning for Music Feature-Informed Captioning. In *Proceedings of the 6th Conference on AI Music Creativity (AIMC)*, 2025. URL https://arxiv.org/abs/2506.15154.

Xuanming Cui, Jianpeng Cheng, Hong-you Chen, Satya Narayan Shukla, Abhijeet Awasthi, Xichen Pan, Chaitanya Ahuja, Shlok Kumar Mishra, Taipeng Tian, Qi Guo, Ser-Nam Lim, Aashu Singh, and Xiangjun Fan. Think Then Embed: Generative Context Improves Multimodal Embedding. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=AKXXMK5YTI.

SeungHeon Doh, Keunwoo Choi, Jongpil Lee, and Juhan Nam. LP-MusicCaps: LLM-Based Pseudo Music Captioning. In *Proceedings of the 24th International Society for Music Information Retrieval Conference (ISMIR)*, 2023. URL https://arxiv.org/abs/2307.16372.

SeungHeon Doh, Minhee Lee, Dasaem Jeong, and Juhan Nam. Enriching Music Descriptions with A Finetuned-LLM and Metadata for Text-to-Music Retrieval. In *ICASSP 2024 - 2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 826–830. IEEE, 2024. doi: 10.1109/icassp48485.2024.10446380. URL http://dx.doi.org/10.1109/ICASSP48485.2024.10446380.

Gemini Team. Gemini 2.5: Pushing the Frontier with Advanced Reasoning, Multimodality, Long Context, and Next Generation Agentic Capabilities. arXiv:2507.06261, 2025. URL https://arxiv.org/abs/2507.06261.

Sreyan Ghosh, Arushi Goel, Lasha Koroshinadze, Sang-gil Lee, Zhifeng Kong, Joao Felipe Santos, Ramani Duraiswami, Dinesh Manocha, Wei Ping, Mohammad Shoeybi, and Bryan Catanzaro. Music Flamingo: Scaling Music Understanding in Audio Language Models. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://openreview.net/forum?id=RS7T9S16Bl.

HeartMuLa Team. HeartMuLa: A Family of Open Sourced Music Foundation Models. arXiv:2601.10547, 2026. URL https://arxiv.org/abs/2601.10547. HeartCLAP: see Section 4.

Mojtaba Heydari, Frank Cwitkowitz, and Zhiyao Duan. BeatNet: CRNN and Particle Filtering for Online Joint Beat Downbeat and Meter Tracking. In *Proceedings of the 22nd International Society for Music Information Retrieval Conference (ISMIR)*, 2021. URL https://arxiv.org/abs/2108.03576.

Qingqing Huang, Aren Jansen, Joonseok Lee, Ravi Ganti, Judith Yue Li, and Daniel P. W. Ellis. MuLan: A Joint Embedding of Music Audio and Natural Language. In *Proceedings of the 23rd International Society for Music Information Retrieval Conference (ISMIR)*, 2022. URL https://arxiv.org/abs/2208.12415.

Zhibin Lan, Liqiang Niu, Fandong Meng, Jie Zhou, and Jinsong Su. UME-R1: Exploring Reasoning-Driven Generative Multimodal Embeddings. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://arxiv.org/abs/2511.00405.

Edith Law, Kris West, Michael I. Mandel, Mert Bay, and J. Stephen Downie. Evaluation of Algorithms Using Games: The Case of Music Tagging. In *Proceedings of the 10th International Society for Music Information Retrieval Conference (ISMIR)*, pp. 387–392, 2009. doi: 10.5281/zenodo.1417647. URL https://zenodo.org/records/1417647.

Xiquan Li, Aurian Quelennec, and Slim Essid. TinyMU: A Compact Audio-Language Model for Music Understanding. In *ICASSP 2026 - 2026 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 15707–15711, 2026. doi: 10.1109/icassp55912.2026.11464280. URL https://doi.org/10.1109/icassp55912.2026.11464280.

Ilaria Manco, Benno Weck, SeungHeon Doh, Minz Won, Yixiao Zhang, Dmitry Bogdanov, Yusong Wu, Ke Chen, Philip Tovstogan, Emmanouil Benetos, Elio Quinton, György Fazekas, and Juhan Nam. The Song Describer Dataset: a Corpus of Audio Captions for Music-and-Language Evaluation. In *NeurIPS Workshop on Machine Learning for Audio*, 2023. URL https://arxiv.org/abs/2311.10057.

Matthias Mauch and Simon Dixon. Approximate Note Transcription for the Improved Identification of Difficult Chords. In *Proceedings of the 11th International Society for Music Information Retrieval Conference (ISMIR)*, 2010. URL https://www.isophonics.net/nnls-chroma. Method underlying Chordino and NNLS Chroma.

Jan Melechovsky, Zixun Guo, Deepanway Ghosal, Navonil Majumder, Dorien Herremans, and Soujanya Poria. Mustango: Toward Controllable Text-to-Music Generation. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pp. 8293–8316. Association for Computational Linguistics, 2024. doi: 10.18653/v1/2024.naacl-long.459. URL https://aclanthology.org/2024.naacl-long.459/.

Multimodal Art Projection. MERT-v2-FullSong (Hugging Face Model Card), 2026. URL https://huggingface.co/m-a-p/MERT-v2-FullSong. Model card; a technical report had not been published at the time of writing.

OpenMOSS Team. MOSS-Music Technical Report, 2026a. URL https://github.com/OpenMOSS/MOSS-Music. Project repository: https://github.com/OpenMOSS/MOSS-Music.

OpenMOSS Team. MOSS-Audio Technical Report. arXiv:2606.01802, 2026b. URL https://arxiv.org/abs/2606.01802.

Igor André Pegoraro Santana, Fabio Pinhelli, Juliano Donini, Leonardo Catharin, Rafael Biazus Mangolin, Yandre Maldonado e Gomes da Costa, Valéria Delisandra Feltrim, and Marcos Aurélio Domingues. Music4All: A New Music Database and Its Applications. In *2020 International Conference on Systems, Signals and Image Processing (IWSSIP)*, pp. 399–404, 2020. doi: 10.1109/iwssip48289.2020.9145170. URL https://doi.org/10.1109/iwssip48289.2020.9145170.

Qwen Team. Qwen3.6-27B, 2026. URL https://qwen.ai/blog?id=qwen3.6-27b&lang=en. Official model blog; accessed 26 September 2026.

Abhinaba Roy, Renhang Liu, Tongyu Lu, and Dorien Herremans. JamendoMaxCaps: A Large Scale Music-caption Dataset with Imputed Metadata. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*, 2025. doi: 10.1109/IJCNN64981.2025.11228034. URL https://doi.org/10.1109/IJCNN64981.2025.11228034.

Xian Shi, Xiong Wang, Zhifang Guo, Yongqi Wang, Pei Zhang, Xinyu Zhang, Zishan Guo, Hongkun Hao, Yu Xi, Baosong Yang, Jin Xu, Jingren Zhou, and Junyang Lin. Qwen3-ASR Technical Report. arXiv:2601.21337, 2026. URL https://arxiv.org/abs/2601.21337.

Changli Tang, Qinfan Xiao, Ke Mei, Tianyi Wang, Fengyun Rao, and Chao Zhang. WAVE: Learning Unified & Versatile Audio-Visual Embeddings with Multimodal LLM. In *International Conference on Learning Representations (ICLR)*, 2026. URL https://arxiv.org/abs/2509.21990.

G. Tzanetakis and P. Cook. Musical genre classification of audio signals. *IEEE Transactions on Speech and Audio Processing*, 10(5):293–302, 2002. doi: 10.1109/tsa.2002.800560. URL https://doi.org/10.1109/tsa.2002.800560.

Benno Weck, Holger Kirchhoff, Peter Grosche, and Xavier Serra. WikiMuTe: A Web-Sourced Dataset of Semantic Descriptions for Music Audio. In *MultiMedia Modeling (MMM)*, pp. 42–56. Springer Nature Switzerland, 2024. doi: 10.1007/978-3-031-56435-2_4. URL http://dx.doi.org/10.1007/978-3-031-56435-2_4.

Junda Wu, Warren Li, Zachary Novack, Amit Namburi, Carol Chen, and Julian McAuley. CoLLAP: Contrastive Long-form Language-Audio Pretraining with Musical Temporal Structure Augmentation. In *ICASSP 2025 - 2025 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5. IEEE, 2025a. doi: 10.1109/icassp49660.2025.10888307. URL http://dx.doi.org/10.1109/ICASSP49660.2025.10888307.

Peixi Wu, Biao Yang, Feipeng Ma, Bosong Chai, Bo Lin, Wei Yuan, Fan Yang, Tingting Gao, Hebei Li, and Xiaoyan Sun. LaME: Learning to Think in Latent Space for Multimodal Embedding via Information Bottleneck. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2026. URL https://arxiv.org/abs/2606.13061. Accepted for publication.

Shangda Wu, Guo Zhancheng, Ruibin Yuan, Junyan Jiang, SeungHeon Doh, Gus Xia, Juhan Nam, Xiaobing Li, Feng Yu, and Maosong Sun. CLaMP 3: Universal music information retrieval across unaligned modalities and unseen languages. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 2605–2625. Association for Computational Linguistics, 2025b. doi: 10.18653/v1/2025.findings-acl.133. URL https://aclanthology.org/2025.findings-acl.133/.

Yusong Wu, Ke Chen, Tianyu Zhang, Yuchen Hui, Taylor Berg-Kirkpatrick, and Shlomo Dubnov. Large-Scale Contrastive Language-Audio Pretraining with Feature Fusion and Keyword-to-Caption Augmentation. In *ICASSP 2023 - 2023 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1–5, 2023. doi: 10.1109/icassp49357.2023.10095969. URL https://doi.org/10.1109/icassp49357.2023.10095969.

wx9songs. MOSS-Music-Data-Pipeline, 2026. URL https://github.com/wx9songs/MOSS-Music-Data-Pipeline. Data-pipeline software repository; accessed 25 September 2026.

Xiaomi MiMo Team. MiMo-Audio: Audio Language Models are Few-Shot Learners. arXiv:2512.23808, 2025. URL https://arxiv.org/abs/2512.23808.

Jilan Xu, Carl Thomé, Danijela Horak, Weidi Xie, and Andrew Zisserman. Scaling Audio-Text Retrieval with Multimodal Large Language Models. arXiv:2602.18010, 2026. URL https://arxiv.org/abs/2602.18010. AuroLA.

Jin Xu, Zhifang Guo, Hangrui Hu, Yunfei Chu, Xiong Wang, Jinzheng He, Yuxuan Wang, Xian Shi, Ting He, Xinfa Zhu, Yuanjun Lv, Yongqi Wang, Dake Guo, He Wang, Linhan Ma, Pei Zhang, Xinyu Zhang, Hongkun Hao, Zishan Guo, Baosong Yang, Bin Zhang, Ziyang Ma, Xipin Wei, Shuai Bai, Keqin Chen, Xuejing Liu, Peng Wang, Mingkun Yang, Dayiheng Liu, Xingzhang Ren, Bo Zheng, Rui Men, Fan Zhou, Bowen Yu, Jianxin Yang, Le Yu, Jingren Zhou, and Junyang Lin. Qwen3-Omni Technical Report. arXiv:2509.17765, 2025. URL https://arxiv.org/abs/2509.17765.

HaeJun Yoo, Yongseop Shin, Insung Lee, Myoung-Wan Koo, and Du-Seong Chang. Omni-Embed-Audio: Leveraging Multimodal LLMs for Robust Audio-Text Retrieval. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 22664–22680. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.1038. URL https://aclanthology.org/2026.acl-long.1038/.

Zuyao You, Zhesong Yu, Mingyu Liu, Bilei Zhu, Yuan Wan, and Zuxuan Wu. GaMMA: Towards Joint Global-Temporal Music Understanding in Large Multimodal Models. arXiv:2605.00371, 2026. URL https://arxiv.org/abs/2605.00371.

Ruibin Yuan, Yinghao Ma, Yizhi Li, Ge Zhang, Xingran Chen, Hanzhi Yin, Le Zhuo, Yiqi Liu, Jiawen Huang, Zeyue Tian, Binyue Deng, Ningzhi Wang, Chenghua Lin, Emmanouil Benetos, Anton Ragni, Norbert Gyenge, Roger Dannenberg, Wenhu Chen, Gus Xia, Wei Xue, Si Liu, Shi Wang, Ruibo Liu, Yike Guo, and Jie Fu. MARBLE: Music Audio Representation Benchmark for Universal Evaluation. *Advances in Neural Information Processing Systems*, 36:39626–39647, 2023. URL https://arxiv.org/abs/2306.10548.

Haina Zhu, Yizhi Zhou, Hangting Chen, Jianwei Yu, Ziyang Ma, Rongzhi Gu, Yi Luo, Wei Tan, and Xie Chen. MuQ: Self-Supervised Music Representation Learning with Mel Residual Vector Quantization. *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, 33, 2025. doi: 10.1109/TASLP.2025.3602320. URL https://doi.org/10.1109/TASLP.2025.3602320.

APPENDIX

A DATA, PROMPTS AND EVALUATION EXAMPLES

A.1 TRAINING DATA FORMAT EXAMPLE: LET IT BE

We use Let It Be to illustrate how one piece of music is organised into structured metadata and multi-view retrieval texts. The metadata below come from the existing consolidated record for this audio. For clarity, we omit song identity fields, auxiliary analysis fields and the lyrics themselves. The song title identifies the example only and does not enter retrieval texts.

Table 4: Ten English retrieval views, with one candidate per view. † marks the seven strong-positive types; final training selects one text per song.

View	Expressive focus
Core tags	Core musical concepts
Single attribute: theory	One music-theoretic attribute
Single attribute: timbre	One timbral, instrumental or vocal attribute
Composed: theory + timbre†	Music theory and timbre
Composed: cross-dimension†	Genre, vocals, instrumentation and mood
Short caption: style / arrangement†	Style and arrangement
Short caption: vocal / dynamics†	Vocal and accompaniment delivery
Long caption†	Comprehensive full-song description
Natural retrieval query†	General user search intention
Producer retrieval query†	Specialist attribute combinations for producers


```
{
  "genre": [
    "Piano Rock",
    "Gospel Rock",
    "Classic Rock",
    "Soft Rock"
  ],
  "tempo": {
    "bpm": "74",
    "timeSignature": "4/4"
  },
  "vocal": {
    "vocalIdentity": "Male",
    "vocalType": "Gentle close-mic warm mid-range narrative at the
    start, builds to an emotional, slightly raspy, powerful belting
    delivery with natural vibrato and ad-libs"
  },
  "instrument": [
    "Steady Chordal Piano",
    "Warm Round Electric Bass (legato)",
    "Punchy Live Drums (entering later, natural room sound)",
    "Warm Gospel Organ Swells",
    "Melodic Overdriven Guitar Solo (with bluesy bends)",
    "Subtle Shaker Percussion",
    "Clean Arpeggiated Rhythm Guitar"
  ],
  "melody": {
    "key": "C",
    "mode": "Major",
    "chord": [
      "C",
      "G",
      "Am",
      "F",
```

```
      "C",
      "G",
      "F",
      "C"
    ]
  },
  "mood_and_scene": [
    "Hopeful",
    "Reassuring",
    "Anthemic build-up",
    "Live band feel",
    "Uplifting",
    "Nostalgic",
    "Introspective",
    "Cathartic",
    "Spiritual"
  ],
  "lyrics": {
    "language": "en"
  }
}
```


The following ten views follow our protocol and the record above to illustrate the form of training candidates. They are appendix examples, rather than original generated outputs from the frozen training set. The texts share the same musical attributes but focus respectively on concepts, single attributes, attribute combinations, comprehensive descriptions and user retrieval intentions.

Table 5: Ten illustrative retrieval views for Let It Be.

View	Text
tag_core/core	Piano Rock; Gospel Rock; Classic Rock; Piano; Electric Bass; Live Drums; Organ; Male Vocal; Hopeful; Reassuring
attribute_single/theory	Music centered on C major tonality.
attribute_single/timbre	Warm male vocals with natural vibrato.
attribute_composed/theory_plus_timbre	A C major arrangement brings together chordal piano, warm organ, and expressive male vocals with natural vibrato.
attribute_composed/cross_dimension	Piano rock and gospel rock combine in a hopeful, reassuring atmosphere shaped by piano and warm organ.
caption_short/style_arrangement	A hopeful piano rock arrangement combines chordal piano, warm electric bass, live drums, and organ, with expressive male vocals at its center.
caption_short/vocal_dynamics	Warm male vocals with natural vibrato and emotional delivery shape a reassuring gospel rock sound, supported by piano and an uplifting atmosphere.
caption_long/balanced	Centered on C major at around 74 BPM in 4/4 time, the music combines piano rock, gospel rock, and classic rock. Chordal piano, electric bass, live drums, and warm organ form the core arrangement. The prominent harmony includes C, G, Am, and F. Male vocals feature a warm mid-range tone, natural vibrato, and emotional delivery. The overall character is hopeful and reassuring, with an uplifting, nostalgic quality.
retrieval_query/natural	Find piano rock songs with a hopeful and reassuring mood for uplifting listening.
retrieval_query/producer	Search for piano-led music around 74 BPM in C major.

Each view also records its type, granularity and source fields. For example, the producer query corresponds to:


```
{
```

```
  "view_type": "retrieval_query",
  "variant": "producer",
  "used_fields": ["bpm", "key", "mode", "instruments"],
  "granularity": "query",
  "positive_strength": 0.9,
  "text": "Search for piano-led music around 74 BPM in C major."
}
```

Candidate construction includes ten views. Final contrastive training selects one text per song from seven strong-positive types: two combined-attribute views, two short captions, one long caption and two queries. Tag and single-attribute views support fine-grained diagnosis. The positive_strength field denotes a candidate's semantic strength and does not serve as a weighting coefficient in multi-view evaluation.

A.2 Data Construction Pipeline and Prompts

Data construction proceeds through audio metadata extraction, multi-source cleaning and consolidation, and multi-view text generation. The semantic branch extracts genre, vocals, instrumentation, mood and lyrics; the MIR branch estimates tempo, time signature, key and chords. After consolidation, both types of information jointly constrain the musical facts in retrieval texts.

Below we provide edited English versions of the three-stage prompts, with consistent variable names, output fields and formatting instructions. These templates describe the inputs, outputs and constraints at each stage; placeholders take the audio, metadata from each source or the consolidated structured record, as appropriate. The historical V5 production prompt permits non-conflicting style_tags as supplements. This appendix follows a simplified schema that retains only structured attributes and does not use auxiliary captions or the lyrics themselves to generate retrieval texts.

A.2.1 Audio Metadata Extraction

The primary annotator and checkers use the same field system. Unknown scalar values remain empty strings and unknown collections remain empty arrays. Audio provides the evidence for attribute decisions; song titles, filenames and knowledge of existing works do not supply missing content. The output contains seven groups of musical information.

```
# ROLE

You are an expert in music understanding, music information retrieval,
    and
audio-grounded metadata extraction.

# TASK

Analyze the attached music audio once and return a compact, structured
description of all audible and reliably inferable properties needed by
    the
downstream retrieval-data pipeline.

Do not produce style_tags, captions, analysis text, explanations, or
    any field
outside the required JSON schema.

# CORE PRINCIPLES

1. Use only evidence supported by the audio.
2. Do not use the file name, track ID, presumed song identity, artist
    knowledge,
   external knowledge, or memorized metadata as evidence.
3. If a field cannot be determined reliably, keep its required empty
    value.
4. Do not infer mood from BPM alone.
```

```
5. Do not infer key, mode, chords, lyrics, instruments, or vocal
    identity when
   the evidence is ambiguous.
6. Preserve the original language of the lyrics. Do not translate
    lyrics.
7. Output exactly one valid JSON object and no surrounding prose or
    Markdown.

# REQUIRED OUTPUT SCHEMA

{
  "genre": [],
  "tempo": {
    "bpm": "",
    "timeSignature": ""
  },
  "vocal": {
    "vocalIdentity": "",
    "vocalType": ""
  },
  "instrument": [],
  "melody": {
    "key": "",
    "mode": "",
    "chord": []
  },
  "mood_and_scene": [],
  "lyrics": {
    "language": "",
    "content": ""
  }
}

All keys shown above are mandatory. Use "" for an unknown scalar string
     and []
for an unknown list. Never output null.

# FIELD RULES

## 1. genre

- Return at most three principal genres or clearly audible subgenres.
- Prefer standard genre names over vague adjectives.
- Do not infer country, era, or cultural origin unless directly
    supported by
  unmistakable musical evidence.
- Avoid redundant parent-child labels unless both are useful.

## 2. tempo

- bpm must be an integer encoded as a string, for example "118".
- Populate bpm only when the pulse is sufficiently stable and
    identifiable.
- Account for possible half-time and double-time interpretations. If no
     single
  interpretation is reliable, return "".
- timeSignature must use a conventional value such as "4/4", "3/4", or
    "6/8".
- If the meter is ambiguous, return "".

## 3. vocal

- vocalIdentity must be one of: "Male", "Female", "Mixed", "Group",
  "Instrumental", or "".
- Use "Instrumental" only when no sung or spoken vocal is present.
```

```
- vocalType must be one concise English phrase describing only clearly
    audible
  properties, chosen from relevant axes such as timbre, range,
    technique,
  recording distance, dynamics, and emotional delivery.
- Examples of useful attributes include breathy, clear, raspy, close-
    miked,
  falsetto, belted, restrained, layered harmonies, and spoken delivery.
- Do not infer singer identity, age, or biography.

## 4. instrument

- Return up to eight principal audible instruments or production
    elements;
  three to eight is preferred when enough evidence is available.
- Each item should combine a base instrument with an audible timbral,
  performance, or arrangement role when reliable, for example:
  "Punchy Live Drums", "Warm Legato Bass Guitar", or
  "Arpeggiated Acoustic Guitar".
- Do not list an instrument that is merely plausible for the genre.
- Do not split one sound into several near-duplicate entries.

## 5. melody

- key contains only the tonic, such as "C", "F#", or "Bb".
- mode contains a conventional mode such as "major" or "minor".
- Fill key and mode only when both are reasonably reliable; otherwise
    leave the
  uncertain field empty.
- chord is an ordered list of chord symbols supported by the audio.
- Do not invent a chord sequence. If chord recognition is unreliable,
    return [].
- Use standard chord spellings and preserve inversions only when
    clearly
  supported.

## 6. mood_and_scene

- Return up to four concise English phrases describing audible
    emotional
  character or plausible listening context; two to four is preferred
    when
  enough evidence is available.
- Prefer acoustic evidence such as melancholic, tense, uplifting,
    intimate,
  cinematic, dancefloor, or meditative.
- Do not infer mood from BPM alone and do not invent a narrative.

## 7. lyrics

- language must be a concise language code such as "zh", "en", "ja", "
    ko",
  "es", or "mix". Use "" when no reliable language can be identified.
- content must contain only lyrics that are clearly intelligible in the
     audio.
- Preserve the original language and line order. Do not translate,
    paraphrase,
  complete familiar lyrics from memory, or guess inaudible words.
- Use only these section labels when the structure is clear:
  [Intro], [Verse], [Pre-Chorus], [Chorus], [Post-Chorus], [Bridge], [
    Drop],
  [Break], [Instrumental], [Outro].
- If the structure is uncertain, use [Verse] rather than inventing a
    chorus.
- A section label may be followed by one concise Chinese full-width
```

```
  parenthetical arrangement note of 8-24 Chinese characters, but only
    when the
  described instrumentation, dynamics, or ambience is directly audible.
- Optional line-level controls must appear before a lyric line and use
    at most
  one group selected from:
  (slowly), (gradually), (suddenly), (softly), (gently), (restrained),
  (emotional), (powerful), (breathy), (airy), (falsetto), (belt), (
    whisper),
  (vibrato), (spoken), (ad-lib), (pause), (half-time), (double-time).
- Do not add timestamps, translations, commentary, Markdown, or
    unsupported
  section labels.
- For instrumental music or unintelligible singing, return
  {"language": "", "content": ""}.

# FINAL VALIDATION

Before responding, verify silently that:

1. The output has exactly the seven required top-level keys.
2. style_tags, captions, analysis_text_allm, explanations, and
    confidence
   scores are absent.
3. Every non-empty value is grounded in the attached audio.
4. The lyrics are not completed from memory.
5. The JSON is syntactically valid and directly parseable.

Return the JSON object only.
```

A.2.2 Consolidation and Cleaning

Consolidation anchors on the primary prediction, first normalising fields, synonymous tags and key notation, and then verifying across sources. Two runs of the same checker first collapse into one effective opinion, preventing repeated sampling from increasing its voting weight. Corrections or gap filling occur only when two effective checkers agree on an alternative or missing value.

Tempo comparison allows half-time and double-time relationships, with a folding tolerance of 5 BPM and a relative tolerance of 6% within the same tempo cluster. Key comparison normalises enharmonic notation; MIR key estimates may take precedence when key strength reaches 0.5. Chords use available MIR outputs, otherwise retaining supported model results. These rules control cross-source consistency; the human audit in Appendix A.5 measures final annotation quality.

```
# ROLE

You are a senior music information retrieval expert and a conservative
multi-source metadata consolidation engine.

# GOAL

Produce one auditable music_meta object for a track from:

1. one PRIMARY prediction from Gemini 2.5 Pro;
2. two runs from Checker A, Qwen3 Captioner;
3. two runs from Checker B, Qwen3 Instruct; and
4. optional objective MIR evidence.

PRIMARY is the default anchor. CHECKERS are used only to detect clear
hallucinations and fill clear gaps. MIR is used only for objective
    acoustic
fields.
```

```
# INPUT

track_meta:
  id:     {{track_id}}
  title:  {{title}}
  artist: {{artist}}

predictions:
  primary:
    {{music_meta_primary}}

  checker_A_runs:
    - {{music_meta_checkerA_run1}}
    - {{music_meta_checkerA_run2}}

  checker_B_runs:
    - {{music_meta_checkerB_run1}}
    - {{music_meta_checkerB_run2}}

mir_evidence:
  {
    "bpm": null,
    "time_signature": null,
    "key": null,
    "mode": null,
    "key_strength": null,
    "chord_progression": [],
    "pitch_median_hz": null
  }

# MANDATORY PRE-PROCESSING

First collapse the two runs from each checker model. Never count two
    runs from
the same model as two independent votes.

For Checker A and Checker B separately:

1. Numeric and categorical fields
   - Fields: bpm, timeSignature, key, mode, vocalIdentity, lyrics.
    language.
   - If both runs agree after normalization, the checker casts one vote.

   - If the two runs disagree, the checker casts no vote for that field.


2. Set-valued fields
   - Fields: genre, instrument, mood_and_scene.
   - Normalize synonyms and keep only the semantic intersection of the
    two
      runs.
   - Drop an item that appears in only one run as intra-model sampling
    noise.

3. Free-text fields
   - Fields: vocalType and lyrics.content.
   - Extract claims or attributes from each run and retain only content
      supported by both runs.
   - Do not concatenate the two runs.

After this step, there are at most two effective checker votes per
    field:
Checker A and Checker B.

# CORE ANCHOR-THEN-CORRECT POLICY
```

```
Apply the following sequence independently to every field:

1. KEEP
   Start from PRIMARY's value.

2. VERIFY
   - If both effective checkers support PRIMARY, keep it and mark "
    verified".
   - If one checker supports PRIMARY and the other is empty or no-vote,
     keep it
     and mark "verified".
   - If one checker disagrees and the other is empty or no-vote, keep
    PRIMARY
     and mark "weak-disagreement".
   - If both checkers reject PRIMARY and agree on the same alternative,
     replace
     PRIMARY and mark "checker-overridden".
   - If both checkers reject PRIMARY but disagree with each other, keep
     PRIMARY
     for descriptive free text. For objective fields use valid MIR
    evidence;
     otherwise empty the field and mark "conflict-emptied".

3. FILL
   If PRIMARY is empty, fill the field only when both effective
    checkers agree
   on the same value. Mark "gap-filled". Otherwise keep the required
    empty
   value.

4. TRIM
   Normalize duplicates and enforce field-specific caps. Never
    introduce a
   fact absent from every source.

# FIELD-SPECIFIC RULES

## tempo.bpm

- Parse all valid positive BPM values.
- Treat half-time and double-time equivalents as one family when the
    folded
  values differ by no more than 5 BPM.
- Use a relative agreement tolerance of 6 percent within the selected
    family.
- If MIR conflicts with PRIMARY and at least one effective checker
    supports the
  MIR family, use the MIR-aligned cluster median.
- Return a rounded integer as a string.

## tempo.timeSignature

- Normalize equivalent formatting such as "4 / 4" to "4/4".
- Use the majority among PRIMARY and the two effective checkers.
- PRIMARY wins a tie.
- Valid MIR meter may resolve an otherwise unresolved conflict.

## melody.key and melody.mode

- Normalize enharmonic spellings, for example C# and Db.
- Split combined values such as "C minor" into key="C" and mode="minor".

- Override non-empty PRIMARY only when both effective checkers agree on
    the
```

```
  alternative and MIR supports it.
- Valid MIR with key_strength >= 0.5 overrides conflicting model
    estimates.

## melody.chord

- If MIR chord_progression is non-empty, use it directly.
- Otherwise keep a non-empty PRIMARY chord list.
- If PRIMARY is empty, fill only when both effective checkers support a
    common
  sequence with a longest common subsequence of at least four chords.
- Never splice unrelated chord fragments from different sources.
- If the resulting sequence is grossly inconsistent with the final key
    and
  mode, return [].

## genre

- Normalize synonyms and parent-child genre relations.
- Begin with PRIMARY's ordered list.
- Keep a PRIMARY genre when at least one effective checker supports the
    same
  normalized family.
- Remove it only when both effective checkers explicitly support a
    conflicting
  family and neither supports the PRIMARY family.
- Add a missing genre only when both effective checkers support it.
- Return at most five entries.

## mood_and_scene

- Normalize semantically equivalent descriptions into shared clusters.
- Apply the same conservative keep, override, and fill policy as genre.
- Return at most five concise entries.

## vocal.vocalIdentity

- Normalize to "Male", "Female", "Mixed", "Group", "Instrumental", or
    "".
- Override PRIMARY only when both effective checkers agree on a
    different
  value.

## vocal.vocalType

- Compare attributes along timbre, range, technique, recording distance
    ,
  dynamics, and emotion.
- Preserve PRIMARY attributes supported by at least one effective
    checker.
- Remove an attribute only when both effective checkers contradict it
    on the
  same axis.
- Add an attribute only when both effective checkers support it.
- Rewrite the result as one concise English phrase of at most 40 words.

## instrument

- Normalize near-synonyms to instrument families while preserving
    useful
  specificity.
- Begin with PRIMARY's ordered list.
- Keep a PRIMARY item when at least one checker supports a near-
    equivalent.
- Remove it only when both effective checkers clearly contradict it.
```

```
- Add a missing instrument only when both effective checkers support it.

- Deduplicate and order by:
  drums, bass, guitars, keys, strings/brass/woodwinds, traditional
    instruments,
  synths, backing vocals, other.
- Return at most eight entries.

## lyrics.language

- Normalize to concise language codes such as zh, en, ja, ko, es, or
    mix.
- Override PRIMARY only when both effective checkers agree on another
    language.
- Cross-check against the script used in lyrics.content.

## lyrics.content

- Never splice lyric fragments across sources.
- Keep PRIMARY verbatim by default.
- If PRIMARY is empty, use checker lyrics only when both effective
    checkers
  provide near-identical content; choose the more complete version.
- Correct an individual PRIMARY line only when both effective checkers
    support
  the same correction.
- If PRIMARY is empty and checker lyrics diverge substantially, return
    "".
- Never complete lyrics from song identity or external memory.

# STRICT OUTPUT

Return exactly one JSON object with this structure:

{
  "music_meta": {
    "meta": {
      "title": "<copy track_meta.title>",
      "artist": "<copy track_meta.artist>"
    },
    "genre": [],
    "tempo": {
      "bpm": "",
      "timeSignature": ""
    },
    "vocal": {
      "vocalIdentity": "",
      "vocalType": ""
    },
    "instrument": [],
    "melody": {
      "key": "",
      "mode": "",
      "chord": []
    },
    "mood_and_scene": [],
    "lyrics": {
      "language": "",
      "content": ""
    }
  },
  "_diagnostic": {
    "anchor": "primary",
    "effective_checkers": 2,
    "mir_used_fields": [],
```

```
    "field_actions": {
      "genre": "verified",
      "tempo.bpm": "verified",
      "tempo.timeSignature": "verified",
      "vocal.vocalIdentity": "verified",
      "vocal.vocalType": "verified",
      "instrument": "verified",
      "melody.key": "verified",
      "melody.mode": "verified",
      "melody.chord": "verified",
      "mood_and_scene": "verified",
      "lyrics.language": "verified",
      "lyrics.content": "verified"
    },
    "primary_retention_rate": 1.0,
    "notes": ""
  }
}

Each field_actions value must be one of:
"verified", "weak-disagreement", "checker-overridden", "gap-filled",
"conflict-emptied", or "mir-overridden".

# HARD CONSTRAINTS

1. Output one valid JSON object only.
2. Do not output Markdown, comments, reasoning, or surrounding prose.
3. Use "" and [] for empty values. Never use null in the output.
4. Do not output style_tags, captions, or analysis_text_allm.
5. Preserve PRIMARY unless the rules explicitly authorize a change.
6. Never treat two runs from the same checker as two independent votes.
7. Never introduce a fact absent from PRIMARY, both effective checkers,
     and MIR.
8. Copy title and artist from track_meta; do not infer them.
```

A.2.3 Multi-View Text and Atomic Counterfactual Generation

Consolidated metadata become flat fields, from which the generator constructs ten English texts with different semantic emphases. If fewer than four musical dimensions contain valid information, it returns insufficient_facts. Numerical tempo remains a BPM value or a narrow range containing the source value; BPM does not automatically determine mood or fast/slow labels. Chord descriptions remain at song level, and array order does not become a local progression without temporal evidence.

Each atomic counterfactual changes one attribute among BPM, key or mode, vocal identity and dominant instrument, preserving the remaining context and sentence structure. Counterfactuals evaluate attribute discrimination and do not enter the in-batch negatives of standard contrastive training.

```
# ROLE

You are an expert in music understanding, music information retrieval,
    and
text-music representation data construction.

# TASK

Given one flattened music metadata record, generate:

1. exactly ten fact-consistent English positive text views when the
    record has
   sufficient facts; and
2. up to four atomic counterfactual pairs, each changing exactly one
    musical
```

```
   attribute.

# INPUT

{SCHEMA_JSON}

The input may contain:

- id
- audio_path
- fbank_path
- duration_s
- genre
- bpm
- time_signature
- key
- mode
- chords
- vocal_identity
- vocal_style
- instruments
- mood_scene
- lyrics_language
- lyrics_content

Only the following fields are valid generation facts:

genre, bpm, time_signature, key, mode, chords, vocal_identity,
    vocal_style,
instruments, mood_scene.

Use id only to construct output IDs. Never use audio_path, fbank_path,
duration_s, lyrics_language, or lyrics_content as textual evidence. The
     input
does not contain style_tags or analysis captions.

# GLOBAL CONSTRAINTS

1. Use only explicit, non-empty, mutually consistent input facts.
2. All text, positive_text, and negative_text values must be in English.

3. Never mention or infer the track ID, path, file name, dataset, title
    , artist,
   country, era, recording location, producer, or lyric content.
4. Describe song-level properties only. Do not invent local chronology
    such as
   "begins with", "later changes to", "followed by", "resolves to", or
    "ends
   with".
5. Do not create ten superficial paraphrases. Each view must have a
    distinct
   semantic focus and natural wording.
6. Do not infer slow, moderate, fast, Allegro, Andante, or similar
    tempo terms
   from BPM alone.
7. Do not infer energetic, relaxed, driving, or any mood from BPM alone.

8. Output one valid JSON object only, with no Markdown or explanation.

# FACT NORMALIZATION

## Field precedence

Use the flattened structured fields as canonical. There is no
    style_tags
```

```
fallback.

## Value limits

- Round a valid positive bpm to the nearest integer.
- Use time_signature only when it has a valid meter form such as 4/4,
    3/4,
  or 6/8.
- Combine key and mode only when both are valid.
- Select at most three genres, four instruments, two vocal identities,
    three
  vocal-style attributes, and three mood/scene attributes.
- Missing or conflicting fields must be omitted rather than guessed.

## Instruments

- Prefer the base instrument name.
- Remove long parenthetical performance, timbral, or production details
     from
  tag_core, attribute_single, caption_short, and retrieval_query.
- caption_long may retain at most one short, clearly supported
    performance
  detail.
- Do not treat production terms such as "wide stereo" or "large hall
    reverb"
  as separate instruments.

## BPM

- Express BPM only as a number or a narrow numeric interval containing
    the
  source value, for example "around 118 BPM" or "roughly 115-125 BPM".
- Do not shift the center of a range away from the source value.
- Whenever used_fields contains "bpm", the text must contain an
    auditable
  numeric BPM value or interval.

## Chords and scale degrees

- Remove empty, N, none, and no_chord entries.
- For inversions, use the root before the slash for root-frequency
    analysis.
- Mention at most three to five prominent chord names or scale degrees.
- Treat the chord list as song-level evidence. Do not claim that its
    array order
  is a verified temporal progression.
- Derive Roman numerals only when key, mode, and chord roots are all
    reliable.
- If harmonic function is uncertain, use neutral wording such as
  "prominent roots center on scale degrees 1, 4, and 5".
- If key/mode and the chord evidence strongly conflict, mention chord
    names
  without deriving scale degrees.
- If a view uses derived chord degrees, used_fields must contain key,
    mode,
  and chords.

# SUFFICIENT-FACT CHECK

Count these independent semantic dimensions:

1. genre
2. tempo/BPM
3. meter
4. key/mode
```

```
5. chords or chord degrees
6. instruments
7. vocal identity/style
8. mood/scene

If fewer than four dimensions are valid, return:

{
  "id": "<input id>",
  "status": "insufficient_facts",
  "views": [],
  "counterfactuals": []
}

Otherwise set status to "ok" and generate exactly ten views in the
    fixed order
below.

# POSITIVE VIEWS

Every view must contain:

- view_id
- view_type
- variant
- used_fields
- granularity
- positive_strength
- text

used_fields must list exactly the flattened fact fields expressed in
    text and
may contain only:

genre, bpm, time_signature, key, mode, chords, vocal_identity,
    vocal_style,
instruments, mood_scene.

## View 1: tag_core/core

- view_type: "tag_core"
- variant: "core"
- granularity: "concept"
- positive_strength: 0.5
- text: 6-12 semicolon-separated concepts, not a sentence.
- Select from genre, base instrument names, vocal identity/style, and
  mood_scene.
- Do not include an exact BPM or a long chord list.

## View 2: attribute_single/theory

- view_type: "attribute_single"
- variant: "theory"
- granularity: "single_attribute"
- positive_strength: 0.5
- length: 4-16 English words.
- Express exactly one semantic target: BPM, meter, tonality, or
    prominent
  harmony.
- key + mode counts as one tonality target.
- key + mode + chords counts as one chord-degree target.
- Do not combine BPM, meter, and tonality in the same single-attribute
    view.

## View 3: attribute_single/timbre
```

```
- view_type: "attribute_single"
- variant: "timbre"
- granularity: "single_attribute"
- positive_strength: 0.5
- length: 4-16 English words.
- Express exactly one target: one instrument or instrument family, vocal
  identity/style, mood/scene, or genre.

## View 4: attribute_composed/theory_plus_timbre

- view_type: "attribute_composed"
- variant: "theory_plus_timbre"
- granularity: "compositional"
- positive_strength: 0.9
- length: 15-35 English words.
- Combine one reliable theory dimension with one or two timbral or semantic
  dimensions.

## View 5: attribute_composed/cross_dimension

- view_type: "attribute_composed"
- variant: "cross_dimension"
- granularity: "compositional"
- positive_strength: 0.9
- length: 15-35 English words.
- Combine two or three natural retrieval dimensions selected from genre,
  instruments, vocals, and mood.
- Use a different field combination from View 4.
- At least one of Views 4 and 5 must not use genre.

## View 6: caption_short/style_arrangement

- view_type: "caption_short"
- variant: "style_arrangement"
- granularity: "global"
- positive_strength: 1.0
- length: 20-35 English words.
- Emphasize overall style, principal instruments, vocals when available, and
  one or two moods.
- Include at most one theory attribute.

## View 7: caption_short/vocal_dynamics

- view_type: "caption_short"
- variant: "vocal_dynamics"
- granularity: "global"
- positive_strength: 1.0
- length: 20-35 English words.
- When vocals are reliable, center the sentence on vocal identity/style and
  support it with genre, one principal instrument, and mood.
- When vocals are unavailable, do not invent "instrumental"; instead center the
  view on instruments, genre, mood, and meter.
- Use a different information order and syntax from View 6.

## View 8: caption_long/balanced

- view_type: "caption_long"
- variant: "balanced"
```

```
- granularity: "global_detailed"
- positive_strength: 1.0
- length: 60-100 English words.
- Cover all reliable song-level attributes: genre, numeric BPM, meter,
  key/mode, prominent harmony, principal instruments, vocal identity/
    style,
  and mood/scene.
- Omit unavailable fields.
- Use three to five natural sentences, each carrying two or three
    related
  facts.
- Do not always begin with genre.

## View 9: retrieval_query/natural

- view_type: "retrieval_query"
- variant: "natural"
- granularity: "query"
- positive_strength: 0.9
- length: 12-28 English words.
- Express one realistic search intent using one attribute or one common
  two-attribute combination.
- Prefer genre + instrument, genre + vocal, genre + mood, instrument +
    mood,
  or vocal + mood.

## View 10: retrieval_query/producer

- view_type: "retrieval_query"
- variant: "producer"
- granularity: "query"
- positive_strength: 0.9
- length: 10-30 English words.
- Use two or three professional retrieval dimensions.
- Prefer BPM + meter, key/mode + instrument, harmony + genre, BPM +
    instrument,
  or key/mode + vocal.
- Do not default to "I am looking for".
- Views 9 and 10 must use different openings and field combinations.
- At least one query must not use genre.

# ATOMIC COUNTERFACTUALS

Generate at most one pair for each available type:

1. tempo_bpm
2. key_or_mode
3. vocal_identity
4. instrument

Skip a type when the source field is missing or a clean single-
    attribute change
cannot be made.

Every pair must contain:

- pair_id
- counterfactual_type
- changed_field
- original_value
- counterfactual_value
- negative_type, fixed to "single_attribute_hard"
- positive_text
- negative_text
```

```
General atomicity rules:

1. Positive and negative text must have nearly identical syntax, length
    , style,
   and all non-target facts.
2. Change exactly one field.
3. Keep one or two shared context facts.
4. Do not leak the label through negation, awkward grammar, or length.
5. Do not modify genre, mood, instruments, vocal style, or other
    attributes as
   side effects.

## tempo_bpm

- Generate only when bpm is valid.
- The counterfactual BPM must differ by at least 18 BPM.
- Avoid values near half or double the original BPM.
- Both texts must use a number or same-width numeric range.
- Do not couple the change with slow/fast or mood words.
- changed_field: "bpm".

## key_or_mode

- When key and mode are valid, prefer changing mode while keeping key
    fixed.
- Otherwise change key while keeping mode fixed.
- Never change both key and mode.
- Exclude chords from the pair if retaining them would create a
    contradiction.
- changed_field must be "key" or "mode".

## vocal_identity

- Generate only for a clear source identity.
- Prefer Male/Female or Solo/Choir substitutions.
- Do not change a vocal track to Instrumental.
- Keep vocal style, genre, mood, and instruments unchanged.
- changed_field: "vocal_identity".

## instrument

- Generate only when instruments is non-empty.
- Replace exactly one salient instrument with a musically plausible but
     clearly
  different base instrument.
- original_value must copy the selected source item exactly.
- counterfactual_value must be one base-instrument string.
- Use only base instrument names in positive_text and negative_text.
- changed_field: "instruments".

# STRICT OUTPUT SCHEMA

For status="ok", return this exact top-level structure and exactly ten
    view
objects in the specified order:

{
  "id": "<input id>",
  "status": "ok",
  "views": [
    {
      "view_id": "<input id>_tag_core_01",
      "view_type": "tag_core",
      "variant": "core",
      "used_fields": [],
```

```
    "granularity": "concept",
    "positive_strength": 0.5,
    "text": "<English text>"
  },
  {
    "view_id": "<input id>_attribute_single_02",
    "view_type": "attribute_single",
    "variant": "theory",
    "used_fields": [],
    "granularity": "single_attribute",
    "positive_strength": 0.5,
    "text": "<English text>"
  },
  {
    "view_id": "<input id>_attribute_single_03",
    "view_type": "attribute_single",
    "variant": "timbre",
    "used_fields": [],
    "granularity": "single_attribute",
    "positive_strength": 0.5,
    "text": "<English text>"
  },
  {
    "view_id": "<input id>_attribute_composed_04",
    "view_type": "attribute_composed",
    "variant": "theory_plus_timbre",
    "used_fields": [],
    "granularity": "compositional",
    "positive_strength": 0.9,
    "text": "<English text>"
  },
  {
    "view_id": "<input id>_attribute_composed_05",
    "view_type": "attribute_composed",
    "variant": "cross_dimension",
    "used_fields": [],
    "granularity": "compositional",
    "positive_strength": 0.9,
    "text": "<English text>"
  },
  {
    "view_id": "<input id>_caption_short_06",
    "view_type": "caption_short",
    "variant": "style_arrangement",
    "used_fields": [],
    "granularity": "global",
    "positive_strength": 1.0,
    "text": "<English text>"
  },
  {
    "view_id": "<input id>_caption_short_07",
    "view_type": "caption_short",
    "variant": "vocal_dynamics",
    "used_fields": [],
    "granularity": "global",
    "positive_strength": 1.0,
    "text": "<English text>"
  },
  {
    "view_id": "<input id>_caption_long_08",
    "view_type": "caption_long",
    "variant": "balanced",
    "used_fields": [],
    "granularity": "global_detailed",
    "positive_strength": 1.0,
```

```
      "text": "<English text>"
    },
    {
      "view_id": "<input id>_retrieval_query_09",
      "view_type": "retrieval_query",
      "variant": "natural",
      "used_fields": [],
      "granularity": "query",
      "positive_strength": 0.9,
      "text": "<English text>"
    },
    {
      "view_id": "<input id>_retrieval_query_10",
      "view_type": "retrieval_query",
      "variant": "producer",
      "used_fields": [],
      "granularity": "query",
      "positive_strength": 0.9,
      "text": "<English text>"
    }
  ],
  "counterfactuals": [
    {
      "pair_id": "<input id>_cf_tempo_bpm",
      "counterfactual_type": "tempo_bpm",
      "changed_field": "bpm",
      "original_value": 118,
      "counterfactual_value": 148,
      "negative_type": "single_attribute_hard",
      "positive_text": "<English positive text>",
      "negative_text": "<English negative text>"
    }
  ]
}

The counterfactuals array may contain zero to four objects. Do not
    insert the
example tempo pair when bpm is unavailable.

# FINAL SILENT CHECK

Before responding, verify:

1. status is valid;
2. an ok record contains exactly ten views in the required order;
3. every view_id and pair_id uses the exact input id;
4. every used_fields entry is allowed and is actually expressed in text
    ;
5. all generated texts are non-empty English;
6. all word-count limits are satisfied;
7. no unsupported fact, identity clue, lyric, path, or temporal claim
    appears;
8. every counterfactual changes exactly one field; and
9. the response is one complete, directly parseable JSON object.

Return the JSON object only.
```

A.3 Evaluation Resources, Examples and Data Statistics

Table 6 summarises the seven primary tasks and the public evaluation extensions. Retrieval metrics are computed separately within each candidate pool and averaged equally across tasks to obtain the

seven-task mean; multi-view results average equally across views. The following examples detail the four public data extensions.

Table 6: Primary retrieval tasks and evaluation-resource statistics. Audio/text counts denote candidates and queries; CF denotes atomic counterfactual pairs. A dash indicates that the task has no counterfactual evaluation.

Task	Data and descriptions	Audio / text counts	CF pairs	Metric
SDD_orig	Human natural descriptions from SDD	547 / 746	—	mapped T2A R@10
JMC-30s	Official JamendoMaxCaps captions and 30-second audio	10,000 / 10,000	—	Bi-R@10
JMC-fullsong	Full-song audio and extended descriptions for the same songs	10,000 / 10,000	—	Bi-R@10
Music4all-US10K	Unique tag signatures and four-view texts	10,000 / 40,000	—	4-view Macro Bi-R@10
In-house	Held-out in-domain data and multi-view queries	10,000 / 81,687	—	Macro mapped T2A R@10
SDD-MV	Ten-view SDD extension	705 / 7,050	2,488	10-view Macro Bi-R@10; CF accuracy
MTG-Test-MV	Ten-view MTG-Jamendo test extension	1,000 / 10,000	3,327	10-view Macro Bi-R@10; CF accuracy

A.3.1 JMC-30s AND JMC-FULLSONG

Both settings use the same 10,000 songs in the same candidate order. JMC-30s uses the official descriptions, which are generated by Qwen2-Audio, and the designated 30-second audio; JMC-fullsong retains the original captioning task prompt and uses Qwen3-Omni-30B-A3B-Captioner to regenerate descriptions from longer audio. Generation and MUUNE retrieval both use at most the first 300 seconds; shorter recordings use their full decodable duration.

The original prompt for JMC-fullsong is:

```
Generate a detailed caption for this music piece,
primarily focusing on instrumentation, genre,
mood, rhythm, and scenario.
```

All 10,000 songs have valid captions. In English word tokens, mean/median lengths are 62.4/61 words for JMC-30s and 453.8/437 words for JMC-fullsong. Word counts treat forms joined by internal hyphens or apostrophes as single words.

Below we compare the two original descriptions of song 949455.

JMC-30s, 0–30 seconds:

```
The music piece is an instrumental electronic track with a synth pop
    genre, characterized by
a dreamy mood and played at a tempo of 72.3 bpm in a 4/4 time signature.
     It evokes a slow
dance scene in a movie.
```

JMC-fullsong, 0–min(300 seconds, full audio duration):

```
The audio clip is a high-fidelity, professionally produced trance track
    , emblematic of the
genres peak in the late 1990s and early 2000s. The piece unfolds with a
     richly layered,
atmospheric soundscape: a crystalline, arpeggiated synthesizer melody
    dances atop a deep,
resonant bassline and a lush pad, creating a sense of anticipation and
    vastness. The rhythm
is driven by a steady four-on-the-floor kick drum, crisp hi-hats, and a
     syncopated, rolling
bassline, all tightly locked in a classic trance groove. As the track
    progresses, a
sweeping, cinematic lead synth emerges, delivering an uplifting and
    emotionally resonant
motif that is both melodic and anthemic.

The arrangement is meticulously structured, following the conventions
    of trance music: it
begins with an atmospheric build-up, transitions into a euphoric main
    section with a full
melodic and harmonic focus, and features a dramatic breakdown where the
     beat drops out,
leaving the lead melody and pads suspended in a vast, digital space.
    The track then climaxes
with a powerful, energetic return to the main theme, marked by lush
    synth textures and a
driving rhythm. The sound design is pristine, with wide stereo imaging,
     subtle use of delay
and reverb, and a polished, club-ready mix.

The mood is intensely uplifting and euphoric, evoking feelings of hope,
     transcendence, and
communal celebration. The track is designed for large-scale dance
    environmentsclubs,
festivals, and raveswhere its anthemic melodies and driving rhythm are
    meant to inspire
collective joy and release. The absence of vocals or non-musical
    elements reinforces its
universal, instrumental nature, focusing all attention on the emotional
     journey crafted
through sound alone.

In summary, this audio clip is a quintessential example of late-1990s/
    early-2000s trance,
characterized by its euphoric melodies, driving rhythm, and immersive
    production. It is a
product of the genres golden era, crafted for the dancefloor and the
    emotional high that
defines the trance experience.
```

The 30-second description focuses on overall style, mood and rhythm; the full-song version contains richer descriptions of instrumentation, dynamics and structure. This comparison changes both audio coverage and description source, and therefore reflects end-to-end retrieval under long-audio and long-text conditions. The controlled experiment in Table 3(b) instead fixes queries and the model while varying audio coverage. The texts above preserve the frozen data verbatim; statements such as stylistic era remain generated content and do not constitute additional human-verified facts.

A.3.2 MUSIC4ALL-US10K

We construct a subset with unique tag signatures from tag-rich Music4All records. We normalise and deduplicate genre and tag fields, compute signatures as unordered sets and retain 10,000 distinct

signatures. Each song contains 5–8 normalised tags, with a mean of 5.48. Audio uses the official 30-second clips.

The four text forms are complete canonical tags, a short caption, a natural retrieval query and a producer query. Canonical tags preserve source concepts. Qwen3.6-27B generates short captions and natural queries from tags and Spotify energy, danceability and valence attributes; producer queries combine canonical tags, BPM and key/mode. Source tags for era or region express stylistic associations only, without implying artist identity or recording location.

The following four frozen views correspond to song 7unKKy44ZRZtEgs7.

Table 7: Four frozen text views for Music4all-US10K song 7unKKy44ZRZtEgs7.

View	Text
tag_core/canonical_unique	70s; baroque; brasil; brazilian; dreamy; mpb; progressive; relaxing
caption_short/qwen27b_balanced	A relaxing baroque and MPB track with low energy, moderately danceable rhythm, and balanced valence, evoking a dreamy 70s Brazilian progressive style.
retrieval_query/qwen27b_natural	dreamy 70s brazilian music with low energy and balanced mood
retrieval_query/producer_tag_theory	Find music in D major around 92 BPM, tagged with 70s, baroque, brasil, brazilian, dreamy, mpb, progressive, and relaxing.

Unique tag signatures do not guarantee unique natural-language expressions. The four views contain 10,000, 9,795, 7,320 and 10,000 distinct text groups, respectively; the largest shared natural-query group corresponds to 85 songs. When records within a view share the same normalised text, evaluation includes all corresponding songs in the positive set and uses the rank of the best positive. Duplicate query rows remain, and within-view scores average over sample rows.

A.3.3 SDD-MV: SONG 347879

SDD-MV supplements Song Describer audio with structured attributes and ten-view texts. Of 706 source records, 705 have ten complete, valid views, yielding 7,050 retrieval texts. We illustrate song 347879, whose audio duration is approximately 120 seconds.

Table 8: Consolidated metadata for SDD-MV song 347879.

Attribute	Metadata
Genre	Alternative Rock; Indie Rock; Blues Rock
Tempo and time signature	99 BPM; 4/4
Key	E minor
Vocals	Male; Raspy; Energetic Delivery; Strained Chest Voice
Principal instrumentation	Live Drums; Bass Guitar; Overdriven Electric Guitar; Rhythm Electric Guitar
Mood and scene	Driving; Energetic; Raw; Live Bar Performance

The following excerpts show frozen views at different granularities and preserve their original wording.

Table 9: Selected frozen views for SDD-MV song 347879.

View	Text
tag_core/core	Alternative Rock; Indie Rock; Blues Rock; Live Drums; Bass Guitar; Electric Guitar; Male Vocal; Raspy; Energetic; Raw
attribute_single/theory	Music centered on E minor tonality.
attribute_single/timbre	Raspy male vocals with energetic delivery.
attribute_composed/theory_plus_timbre	An E minor track driven by overdriven electric guitar riffs and crunchy rhythm chords.
caption_short/style_arrangement	A raw alternative rock track featuring punchy live drums, driving bass, and overdriven electric guitars, supported by raspy male vocals and an energetic atmosphere.
retrieval_query/natural	Find alternative rock songs with raspy male vocals and a raw, energetic mood.
retrieval_query/producer	Search for tracks in E minor around 99 BPM with a 4/4 time signature.

The mode counterfactual for the same song is:


```
{
  "counterfactual_type": "key_or_mode",
  "changed_field": "mode",
  "original_value": "minor",
  "counterfactual_value": "major",
  "positive_text": "A track centered on E minor tonality with
    overdriven guitars.",
  "negative_text": "A track centered on E major tonality with
    overdriven guitars."
}
```


The two texts differ only in replacing minor with major. A prediction counts as correct when cosine similarity between the audio and positive text strictly exceeds that for the negative text; ties count as errors. SDD-MV contains 2,488 counterfactual pairs: 705 each for BPM, key/mode and instrument, and 373 for vocal identity.

A.3.4 MTG-TEST-MV

MTG-Test-MV selects 1,000 songs from MTG-Jamendo test data with non-empty genre, instrument and mood tags, balancing tag coverage with artist and album diversity. Audio language models supplement semantic fields, MIR adds music-theoretic attributes, and the same ten-view protocol as SDD-MV generates the texts. All 1,000 songs have complete ten-view annotations.

We illustrate track_1393186. Its consolidated metadata specify Pop Rock/Indie Rock, 100 BPM, 4/4 and C major, with drums, bass and several electric-guitar roles; mood attributes are energetic, optimistic and motivational. Frozen text excerpts follow.

Table 10: Selected frozen views for MTG-Test-MV track_1393186.

View	Text
tag_core/core	Pop Rock; Indie Rock; Instrumental; Live Drums; Bass Guitar; Electric Guitar; Energetic; Optimistic
attribute_single/theory	Music centered on C major tonality.
attribute_composed/theory_plus_timbre	A C major instrumental track built around clean arpeggiated electric guitar and punchy live drums.

Continued on next page

Table 10 (continued)

View	Text
caption_short/style_ arrangement	An energetic Pop Rock instrumental featuring punchy live drums, clean round bass, and distorted power chord guitars with an optimistic drive.
caption_long/balanced	This instrumental Pop Rock and Indie Rock track is set in C major with a tempo around 100 BPM in 4/4 time. The arrangement combines punchy live drums, a clean round bass guitar, and distorted power chord guitars with palm-muted techniques. Clean arpeggiated electric guitar and melodic leads add texture to the driving rhythm. The harmony emphasizes the I, IV, V, and vi degrees, creating a stable and optimistic foundation. The overall mood is energetic and motivational, suitable for active listening.
retrieval_query/natural	Find energetic and optimistic Pop Rock instrumental tracks.
retrieval_query/producer	Search for music around 100 BPM in 4/4 time, key of C major.

The BPM counterfactual for this record is:


```
{
  "counterfactual_type": "tempo_bpm",
  "changed_field": "bpm",
  "original_value": 100,
  "counterfactual_value": 130,
  "positive_text": "Music around 100 BPM in 4/4 time with punchy live
    drums.",
  "negative_text": "Music around 130 BPM in 4/4 time with punchy live
    drums."
}
```


MTG-Test-MV contains 3,327 counterfactual pairs: 1,000 each for BPM, key/mode and instrument, and 327 for vocal identity. Samples without reliable vocal identity do not receive this counterfactual type.

A.3.5 Audio Durations and Coverage at 300 Seconds

Table 11 reports the duration of available evaluation audio before truncation. Full coverage is the fraction of recordings no longer than 300 seconds. Time retained is $\sum_i \min(d_i, 300) / \sum_i d_i$, where d_i is the original duration in seconds. P50, P90 and P95 are linearly interpolated quantiles over recordings.

Table 11: Evaluation audio durations (seconds) and coverage under a 300-second budget. Full coverage and time retained are percentages.

Dataset	Audio	P50	Mean	P90	P95	Max.	Full coverage	Time retained
In-house	10,000	221.1	224.1	297.5	325.2	588.5	90.41	98.01
SDD_orig	547	215.4	221.8	312.7	353.8	666.0	87.20	96.35
SDD-MV	705	217.2	225.6	317.0	363.6	753.7	86.67	95.08
MTG-Test-MV	1,000	212.7	229.8	337.3	404.7	3,931.3	83.80	90.67
JMC-fullsong	10,000	215.2	253.3	421.3	533.5	6,790.6	74.14	81.89
JMC-30s	10,000	30.0	30.0	30.0	30.0	30.0	100.00	100.00
Music4all-US10K	10,000	30.0	30.0	30.0	30.0	30.0	100.00	100.00

The budget fully covers 74.14–90.41% of songs in the full-source evaluations. In-house retains 98.01% of total audio time with 90.41% full coverage. JMC-fullsong has a more pronounced long tail, with 74.14% full coverage and 81.89% time retained. A few very long recordings increase the mean and maximum of MTG-Test-MV and JMC-fullsong, motivating the accompanying quantiles.

For JMC-30s and Music4all-US10K, durations refer to the actual approximately 30-second evaluation clips; 100% coverage does not imply that their original songs are shorter than 300 seconds. JMC-fullsong uses the complete source MP3s for the same 10,000 JMC IDs. SDD_orig and SDD-MV use different candidate pools of 547 and 705 songs, respectively, and are therefore counted separately.

A.4 Data Generation Models, MIR Tools and Field Constraints

A.4.1 Data Generation Models

Table 12: Models and tools used for data generation.

Stage	Model or tool	Output role
Primary semantic annotation	Gemini 2.5 Pro	Structured musical attributes and lyrics
Semantic verification	Qwen3-Omni-30B-A3B-Captioner, Qwen3-Omni-30B-A3B-Instruct	Cross-verification and missing-attribute completion; two runs first consolidate by within-model agreement
Supplementary semantic verification	MiMo-Audio, MOSS-Music	Additional checks of music semantics, vocals and other attributes
MIR attributes	BeatNet, Chordino, Essentia	Tempo, time signature, key and chords
Multi-view text	Qwen3.6-27B	Tags, attribute descriptions, captions, queries and atomic counterfactuals
JMC-fullsong captions	Qwen3-Omni-30B-A3B-Captioner	Full-song caption generation with the original captioning prompt

Descriptive attributes from the primary annotator and checkers, together with MIR acoustic estimates, form weakly supervised metadata. Lyrics support audio-to-music-analysis supervision; verbatim lyrics, song titles, artist names and file paths do not provide evidence for generating multi-view retrieval texts.

A.4.2 MIR Tools

Our MIR branch uses MusicCpuPipeline within MusicToolsPipeline from MOSS-Music-Data-Pipeline (wx9songs, 2026), combining Chordino, BeatNet and Essentia. This combination extracts harmonic, beat and tonal information in parallel.

Table 13: MIR tools, outputs and uses.

Tool	Analysis	Output and use in this work
BeatNet	Beat and downbeat positions; beat indices	Estimates BPM from beat sequences; within-bar beat indices support time-signature representation
Chordino	Chord recognition from pitch-class features	Outputs temporally located chords; we use song-level chord lists and principal chord information
Essentia	Key, mode and melodic pitch analysis	Uses key, scale and strength for tonal normalisation and source arbitration; pitch summaries provide auxiliary analysis

Audio first converts to 16 kHz mono before entering the MIR branch; each tool internally processes the signal according to its analysis requirements. During consolidation, valid BPM values round to integers, key and scale map to tonic and mode, respectively, and chord symbols retain parseable roots and qualities.

Following FIGMA's mapping, our data map BeatNet's maximum beat number to N/4 as the time signature. This is a unified metrical representation and does not cover every compound-metre interpretation. Chord texts express principal chords or sets of scale degrees; a full-song list does not constitute a temporally corrected progression. MIR estimates can contain double-time, relative-key and local chord errors; Appendix A.5 reports human-audited accuracy for tempo, key and time signature.

The upstream project also includes SongFormer structural segmentation, segment-level ASR and GPU-based instrument analysis. We cite its MIR tool combination here; cross-source verification and multi-view retrieval supervision follow our own protocol.

A.4.3 Field Constraints and Text Filtering

Before metadata enter retrieval texts, they undergo structural parsing, field normalisation, source-consistency checking and textual factual verification. Normalised BPM, time signature, key/mode and chords take precedence over auxiliary tags; missing or clearly conflicting values are not invented. Positive texts describe only song-level attributes. Long captions retain reliable music-theoretic and instrumentation information, while natural and producer queries use different attribute combinations.

Generated candidates also store used_fields to link text with metadata fields. Counterfactuals record the target field, original value and replacement value to verify single-attribute changes. Training excludes evaluation records by song ID and audio identifier, then deduplicates normalised text and selects one strong-positive training text per song.

Table 14: Human audit of structured tags and MIR attributes. Rates and accuracies are percentages.

Dataset	Songs	Tags	Tag halluc.	Key acc.	Mode acc.	BPM acc.	Metre acc.
Inhouse	500	7,894	6.25	-	-	-	-
SDD	200	2,729	4.40	91.0	93.5	86.5	87.0
MTG	200	2,672	4.04	91.5	95.0	87.0	83.0

A.5 Human Audit of Annotation Accuracy

We conduct human spot checks of structured tags, MIR attributes and multi-view queries, and also examine SDD's original human descriptions and MTG's official tags. All annotators have substantial musical expertise, most come from music-related educational institutions, and senior experts perform a second verification. Structured-tag checks cover 500 in-domain songs, 200 SDD songs and 200 MTG songs.

Table 15: Human audit of queries and original tags.

Audited content	Source	Count	Hallucination rate
SDD-MV multi-view queries	Pipeline-generated	2,000	2.80%
SDD Official Query	Original dataset	347	3.75%
MTG-Test-MV multi-view queries	Pipeline-generated	2,000	2.55%
MTG original tags	Original dataset	1,311	4.35%

Tag hallucination rates use the number of audited tags as the denominator. BPM assessment permits half-time or double-time equivalence and a deviation of up to ± 5 BPM from the human annotation. Key, mode and time signature each have separate accuracy scores. Dashes indicate items without reported results for that audit group.

Multi-view query hallucination rates are 2.80% for SDD and 2.55% for MTG. Automatic annotations therefore retain residual errors after verification; original public descriptions and tags are also not noise-free. Audit units and sample counts differ across sources, so these results describe source-specific quality rather than statistically significant differences in superiority.

B Implementation Details

B.1 Model Architecture, Parameters and Pooling

We initialise the audio language model from Qwen3-ASR-1.7B and adapt the Music Reasoner through music metadata supervision. In the initialisation ablation, Qwen3 AuT follows the same audio–language design, with its audio encoder initialised from Qwen3-Omni's AuT and Qwen3-1.7B as the language backbone. The MOSS-Music comparison uses an 8B music-language backbone. The Qwen3-ASR instance contains a 24-layer audio Transformer encoder and a 28-layer language decoder with hidden size 2048. Audio converts to 16 kHz mono and 128-bin filter-bank features, passes through convolutional downsampling and audio encoding, and projects into the language space. Audio and text tokens share a causal decoder.

Audio input retains at most the first 300 seconds, text queries contain at most 256 tokens, MMOT contains at most 2048 tokens and the total sequence limit is 8192 tokens. Both retrieval paths output 2048-dimensional L2-normalised vectors and compute similarity by dot product.

EOS-LoRA. We freeze the original parameters and inject LoRA into attention projections q, k, v, o and feed-forward gate, up and down projections in all 28 language-decoder layers:

$$W' = W + \frac{\alpha}{r} BA. \tag{5}$$

Here $r = 370$, $\alpha = 740$ and LoRA dropout is 0.05. Audio and text share the adapted decoder. We read the native EOS token's final-layer hidden state and apply final RMSNorm and L2 normalisation to obtain the representation:

$$z^{\mathrm{EOS}}(x) = \mathrm{norm}_2(R_{\theta,\Delta\theta}(x)_{p_{\mathrm{EOS}}}) \,. \tag{6}$$

Independent EH. Following Cui et al. (2026), we freeze the complete Music Reasoner and copy its final K Transformer layers and final RMSNorm to form a trainable embedding head H_ϕ. Let L denote the decoder depth. Then

$$z^{\mathrm{EH}}(x) = \mathrm{norm}_2\Big[H_\phi\big(\mathrm{stopgrad}\big(R_\theta^{1:L-K}(x)\big)\big)_{p_{\mathrm{EH}}}\Big] \,. \tag{7}$$

The principal setting is $L = 28, K = 8$: the frozen first 20 layers precede the eight copied layers, while the original Reasoner's final eight layers remain available for music analysis generation. Audio and text use the same EH, without an additional output projection layer.

Table 16: Representation schemes, trainable parameters and pooling positions.

Scheme	Trainable parameters	Representation forward pass	Pooling position
EOS-LoRA	403,128,320	28 decoder layers with LoRA	Last valid EOS
EH, 8 layers	402,690,048	20 frozen layers + 8 trainable layers	Final assistant-prefix token, before appending EOS

Trainable parameter counts differ by approximately 0.11%. This comparison controls adaptation parameter scale; the scope of parameter updates and the pooling position remain components of each representation scheme.

Table 17: Trainable parameter counts by EH depth.

EH depth	Trainable parameters
1 layer	50.34M

Continued on next page

Table 17 (continued)

EH depth	Trainable parameters
2 layers	100.67M
4 layers	201.35M
8 layers	402.69M

B.2 Message Structures for the Four Representation Configurations

Below, [AUDIO] denotes expanded audio tokens and [EOS] denotes the tokenizer's native <|im_end|>. Message schematics omit other ChatML delimiters to emphasise task content; `\n` denotes an actual newline. EH reads the final token of the assistant generation prefix, whereas EOS-LoRA reads EOS itself. Neither requires decoding a retrieval word first.

The music analysis instruction is:

```
Analyze this music and infer Genre, Vocal, Instrument, Tempo, Key_mode,
Mood_scene, and Lyrics. Output only in <music_meta> format.
```

Reasoner SFT.

```
system:    [music analysis instruction]
user:      [AUDIO]
assistant: <music_meta> ... seven metadata fields ... </music_meta>[EOS
    ]
```

SFT applies autoregressive generation supervision to metadata and the end token, updating the complete Music Reasoner. Paired retrieval text does not enter this stage's input.

EOS-LoRA / without MMOT.

```
system:    Summarize the above audio in one word for music retrieval:
user:      [AUDIO]
assistant: [EOS]
            ^ pool EOS
```

EOS-LoRA / with MMOT.

```
system:    [music analysis instruction]
user:      [AUDIO]
assistant: <music_meta> ... music analysis ... </music_meta>[EOS]
                                                              ^ pool
    EOS
```

Training jointly applies generation and retrieval objectives to the same sequence. At inference, the adapted model generates music analysis, after which we read the EOS representation at the end of generation. Under causal generation, the last metadata token predicts EOS, while retrieval uses the hidden state of EOS itself.

EH / without MMOT.

```
system:    Summarize the above audio in one word for music retrieval:
user:      [AUDIO]
assistant: \n[EOS]
           ^ pool the final prefix token before EOS
```

EH / with MMOT.

```
system:    [music analysis instruction]
user:      [AUDIO]
assistant: <music_meta> ... generated music analysis ... </music_meta>
user:      Summarize the above music and music metadata in one word for
     retrieval:
assistant: \n[EOS]
           ^ pool the final prefix token before EOS
```

In the EH path with MMOT, the frozen Reasoner first generates music analysis; the original audio, analysis text and retrieval instruction then undergo joint encoding. The second representation forward pass updates EH, while the generation path remains frozen.

The text side consistently uses:

```
system:    Summarize the above sentence in one word for music retrieval
    :
user:      [TEXT QUERY]
assistant: [representation position]
```

EOS-LoRA appends EOS at the end and reads its representation; EH uses the last assistant-prefix position. The phrase 'one word' in the prompt guides semantic aggregation; the retrieval output is a continuous vector.

B.3 Generation and Context Handling

Configurations without MMOT construct retrieval representations directly from audio. Configurations with MMOT use audio-conditioned analysis and do not receive paired retrieval queries during generation. EH's first-pass analysis uses greedy decoding, generates at most 2048 tokens and stops at EOS. Analyses from the same Reasoner can be cached for subsequent index construction or fixed-context evaluation.

Generated content follows the <music_meta>...</music_meta> format and retains seven field groups: Genre, Vocal, Instrument, Tempo, Key_mode, Mood_scene and Lyrics. When content exceeds the length budget, Lyrics is shortened first, preserving other attributes and the closing structure. EH's online processing also restores missing format boundaries. If structured trimming is not possible, token-budget truncation preserves the closing suffix. Format repair introduces no new musical attributes.

We separately record the original stopping state, format repairs and field completeness to distinguish generation termination from post-processing outcomes. Text queries truncate at 256 tokens, the attention mask excludes padding throughout the input, and pooling always targets the intended position within the valid sequence.

B.4 Training Objectives and Configuration

Given audio a, analysis instruction p_r and metadata target $c^\star$, music knowledge adaptation uses autoregressive cross-entropy:

$$\mathcal{L}_{\mathrm{SFT}} = -\frac{1}{|c^\star|}\sum_{k=1}^{|c^\star|}\log p_\theta(c_k^\star \mid a, p_r, c_{<k}^\star)\,. \tag{8}$$

The retrieval stage uses a bidirectional contrastive objective. Let $\{z_i^a, z_i^t\}_{i=1}^B$ denote normalised audio and text vectors in the global batch, $s_{ij} = (z_i^a)^\top z_j^t/\tau$ denote similarity, and $P(i)$ denote the positive set sharing the same song ID:

$$\mathcal{L}_{a\to t} = -\frac{1}{B}\sum_{i=1}^{B}\log\frac{\sum_{j\in P(i)}\exp(s_{ij})}{\sum_{j=1}^{B}\exp(s_{ij})}, \qquad \mathcal{L}_{\mathrm{CL}} = \frac{\mathcal{L}_{a\to t}+\mathcal{L}_{t\to a}}{2}. \tag{9}$$

Cross-device all-gather preserves gradients and forms the global candidate set. Standard training pairs each song with one strong-positive text; handling multiple positives by song ID prevents duplicate songs from becoming false negatives.

To discourage representation degeneration, we add a dimension-wise standard-deviation regulariser over each device's local batch:

$$V(Z) = \frac{1}{d}\sum_{k=1}^{d}\max\Big(0, \gamma - \sqrt{\mathrm{Var}(Z_{\cdot k})+\epsilon}\Big), \qquad \mathcal{L}_{\mathrm{var}} = \frac{V(Z^a)+V(Z^t)}{2}. \tag{10}$$

The overall objective is:

$$\mathcal{L}_{\mathrm{rep}} = \mathcal{L}_{\mathrm{CL}} + \lambda_{\mathrm{var}}\mathcal{L}_{\mathrm{var}} + \mathbf{1}_{\mathrm{EOS+MMOT}}\lambda_{\mathrm{CE}}\mathcal{L}_{\mathrm{CE}}. \tag{11}$$

Here $\mathcal{L}_{\mathrm{CE}}$ applies only to EOS-LoRA with MMOT and supervises audio-side metadata and the end token. EH retrieval training optimises the independent copy through the representation loss.

Table 18: Model and loss configuration.

Configuration	Value or setting
Audio sample rate / channels	16 kHz / mono
Audio features	128-bin fbank
Maximum audio duration	300 seconds
Query / MMOT / total sequence limits	256 / 2048 / 8192 tokens
Output dimension / normalisation	2048 / L2
LoRA rank / alpha / dropout	370 / 740 / 0.05
Principal EH depth	8 layers
Contrastive temperature τ	0.02
Standard-deviation threshold γ	0.01
Numerical stability term ϵ	10^{-6}
Variance regularisation weight λ_{var}	25
EOS + MMOT generation weight λ_{CE}	1

C Supplementary Experiments

C.1 Music Representation Transfer on MARBLE

We evaluate MUUNE-EH without MMOT on six MARBLE tasks (Yuan et al., 2023): MagnaTagATune music tagging (Law et al., 2009), GTZAN genre classification (Tzanetakis & Cook, 2002), and MTG-Jamendo instrument, mood/theme, genre and Top-50 tagging (Bogdanov et al., 2019). The experiment freezes the trained audio encoder and learns only task-specific classification probes, testing the transfer of retrieval representations to musical attribute recognition.

C.1.1 Evaluation Setup

Each audio recording converts to mono, and encoding uses at most its first 300 seconds to produce one 2048-dimensional global vector. All tasks use an MLP with one hidden layer:

Table 19: MARBLE transfer results. Paired scores are ROC-AUC/AP; GTZAN reports accuracy. All scores are multiplied by 100.

Model	Params	MTT (R/AP)	GTZAN (Acc.)	MTG Inst. (R/AP)
MUUNE-EH w/o MMOT	2.04B	91.54 / 39.58	86.21	77.97 / 20.77
MERT-Large	330M	90.6 / 37.9	77.6	75.5 / 18.8
Dasheng-1.2B	1.2B	91.5 / 40.4	81.4	75.0 / 19.0
MuQ	310M	90.5 / 38.5	83.8	74.8 / 19.1
MusicFM	330M	90.9 / 38.3	84.1	74.6 / 18.5
AudioMAE++	307M	91.2 / 39.5	80.3	77.1 / 19.9
MATPAC++	307M	90.6 / 38.2	81.4	77.2 / 19.7
A-JEPA	307M	91.0 / 39.2	83.8	76.6 / 19.3
PupuJEPA-Large	307M	91.7 / 40.8	86.9	78.4 / 21.2
PupuJEPA-Huge	632M	91.3 / 39.7	85.9	77.6 / 20.5
MERT-v2-30s	632M	**91.91 / 41.29**	**91.72**	**80.27** / 22.89
MERT-v2-FullSong	632M	91.74 / 41.20	90.69	**80.27 / 23.51**

Model	MTG Mood (R/AP)	MTG Genre (R/AP)	MTG Top-50 (R/AP)
MUUNE-EH w/o MMOT	78.97 / 15.66	**88.41** / 20.68	83.96 / 30.29
MERT-Large	75.3 / 13.5	86.1 / 18.0	82.6 / 29.1
Dasheng-1.2B	76.1 / 15.5	85.5 / 18.8	82.4 / 29.6
MuQ	73.7 / 13.2	85.4 / 19.1	83.0 / 30.2
MusicFM	74.9 / 14.1	85.3 / 19.4	81.9 / 29.7
AudioMAE++	75.6 / 14.0	86.3 / 18.9	83.1 / 31.1
MATPAC++	75.1 / 14.1	85.7 / 19.6	82.5 / 30.2
A-JEPA	74.6 / 14.3	85.5 / 19.2	82.5 / 29.6
PupuJEPA-Large	76.2 / 15.3	86.1 / 20.1	82.8 / 30.5
PupuJEPA-Huge	75.9 / 14.7	85.9 / 20.1	83.1 / 30.7
MERT-v2-30s	**79.44 / 16.68**	88.01 / **21.22**	**84.18 / 32.17**
MERT-v2-FullSong	78.74 / 15.74	87.98 / 20.66	84.13 / 31.62

$$h = \mathrm{Dropout}_{0.2}(\mathrm{ReLU}(W_1 z + b_1)), \qquad \hat{y} = W_2 h + b_2, \tag{12}$$

The hidden layer has width 512, and the output dimension follows the task's label count. Probe training uses random seed 1234 and validation-based epoch selection before evaluating the selected model on the test set. GTZAN reports accuracy; multi-label tasks report macro ROC-AUC and macro average precision (AP).

MUUNE's parameter count is 2.04B for the representation path that actually executes without MMOT: an equally sized EH replaces the original decoder's final eight layers, and the inactive original layers do not count twice. MTG Top-50 uses 32,132/10,888/11,356 training/validation/test recordings; the training split excludes four unavailable recordings.

C.1.2 Results

All scores in the table are multiplied by 100. Paired values denote ROC-AUC/AP; MTT denotes MagnaTagATune.

MUUNE achieves the highest ROC-AUC in the table on MTG genre (88.41), obtains 78.97/15.66 on MTG mood/theme and reaches 86.21 accuracy on GTZAN. Different metrics reveal complementary strengths: for example, MERT-v2-30s (Multimodal Art Projection, 2026) scores higher on MTT, GTZAN and several AP metrics. These results support the retention of transferable musical attribute information in MUUNE retrieval vectors.

MUUNE scores come from the frozen-encoder probe experiment above. Other models retain public results from the comparison report; MERT-v2-FullSong uses task-specific best layers and learning rates from its model card. Differences in pretraining data, model scale and representation selection require interpretation alongside evaluation protocols.

C.2 Embedding Head Capacity

We vary EH depth across one, two, four and eight layers with the SFT-Full Reasoner and CL-Subset contrastive data fixed, using direct encoding without MMOT. The trainable parameter counts are 50.34M, 100.67M, 201.35M and 402.69M, respectively (Table 17). Table 20 reports the retrieval results.

Table 20: Embedding head depth ablation (%). The SFT-Full Reasoner and CL-Subset contrastive data are fixed. Bold marks the best result in each column.

EH depth	SDD orig.	JMC 30s	JMC fullsong	Music4All US10K	In-house	SDD-MV	MTG-Test MV	Mean
1 layer	44.37	6.70	18.74	8.41	46.90	70.25	63.01	36.91
2 layers	46.90	6.99	21.49	9.08	48.90	71.42	63.83	38.37
4 layers	46.51	7.43	21.09	9.48	49.57	72.13	64.68	38.70
8 layers	**51.74**	**8.72**	**24.99**	**12.91**	**54.07**	**76.28**	**68.25**	**42.42**

As EH depth increases from one to eight layers, the seven-task mean rises from 36.91% to 42.42%, a gain of 5.51 percentage points; eight layers achieve the highest result on every task. A deeper representation head provides more capacity for task-relevant information aggregation, motivating eight layers as the principal configuration. One- and two-layer schemes retain strong retrieval performance and offer alternatives when adaptation parameters are constrained.

C.3 Counterfactual Results by Audio Duration

Table 21 complements the retrieval results in Table 3(b). All four limits use the same trained SFT-Full + CL-Full, eight-layer EH without-MMOT model. Query texts, counterfactual pairs and scoring rules remain fixed; no model is retrained. Longer audio improves attribute discrimination, with most of the aggregate gain already attained at 120 seconds.

Table 21: Counterfactual accuracy under fixed-model audio-duration limits (%). The mean gives equal weight to the two tasks. *Default setting used in the main results.

Audio limit	SDD-MV CF	MTG-Test-MV CF	Mean
30 seconds	90.39	88.31	89.35
60 seconds	93.13	91.19	92.16
120 seconds	**93.77**	92.61	93.19
300 seconds*	**93.77**	**93.39**	**93.58**

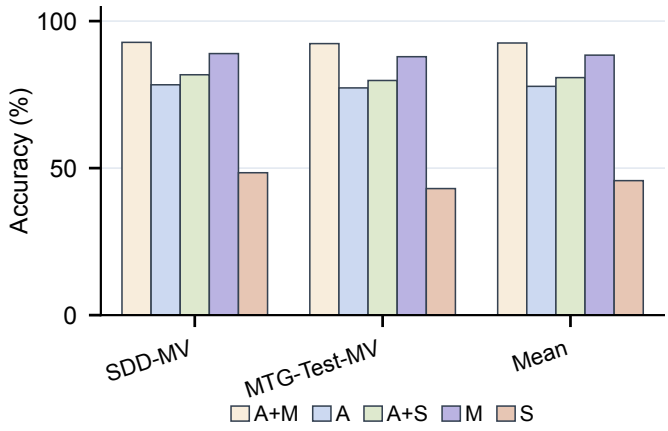


Figure 3: Counterfactual accuracy for the inference-input ablation (%). A+M: audio with matched MMOT; A: audio only; A+S: audio with shuffled MMOT; M: matched MMOT only; S: shuffled MMOT only. This figure accompanies the retrieval results discussed in Section 6 and Table 3(c).