

000 001 002 003 004 005 006 007 008 009 010 011 012 013 014 015 016 017 018 019 020 021 022 023 024 025 026 027 028 029 030 031 032 033 034 035 036 037 038 039 040 041 042 043 044 045 046 047 048 049 050 051 052 053 POLYAUDIO: ADVANCING MULTI-AUDIO REASONING IN LARGE AUDIO LANGUAGE MODELS

005 **Anonymous authors**

006 Paper under double-blind review

ABSTRACT

011 While Large Audio Language Models (LALMs) have achieved superior performance
 012 on reasoning over single audio clips, their ability to understand and reason
 013 over *multiple audio clips* remains a significant challenge. In this paper, we introduce
 014 POLYAUDIO, a novel LALM specifically designed for this complex task. To
 015 systematically train and evaluate our model, we first identify and formalize eleven
 016 foundational multi-audio reasoning capabilities. These capabilities, spanning
 017 sound, music, and speech, are designed to represent a broad range of challenging
 018 real-world scenarios. To enhance these skills, we fine-tune the Qwen2-Audio-7B-
 019 Instruct model using Group Relative Policy Optimization (GRPO). This approach
 020 mitigates common issues associated with Supervised Fine-Tuning (SFT), such as
 021 catastrophic forgetting. Specifically, we construct preference data that explicitly
 022 rewards the model for correctly synthesizing information across multiple audio
 023 clips. Our model, POLYAUDIO, achieves **58.6%** on the MMAU-Pro multi-audio
 024 subset and **71.2%** on our POLYAUDIO-BENCH, substantially outperforming base-
 025 lines on multi-audio reasoning tasks while maintaining its performance on single-
 026 audio tasks. To promote research in this space, we will publicly release the model,
 027 data generators, evaluation scripts, and training recipes at the time of publication.

1 INTRODUCTION

031 Recent state-of-the-art LALMs and multimodal models like Gemini-2.5 (et al., 2025), Qwen-2.5-
 032 Omni (Xu et al., 2025), and Audio Flamingo 3 (Goel et al., 2025) have demonstrated superior au-
 033 dio understanding capabilities well beyond the scope of traditional tasks like Automatic Speech
 034 Recognition (ASR). These models are trained on audio-text question-answer pairs, leveraging nat-
 035 ural language as a more descriptive supervision signal compared to predefined labels for complex,
 036 real-world audio recordings. This has led to impressive performance on a variety of single-audio
 037 tasks, such as spoken language understanding (Tang et al., 2023; Gong et al., 2023), audio caption-
 038 ing (Ghosh et al., 2024a; Deshmukh et al., 2023), and complex reasoning (Ghosh et al., 2024b; 2025;
 039 Goel et al., 2025; Xie et al., 2025). For example, natural language can describe the temporal order of
 040 multiple events (Ghosh et al., 2023) using words such as ‘simultaneous’, ‘before’, and ‘after’, which
 041 helps models learn the intrinsic relationships within an audio stream more effectively. This success,
 042 however, primarily applies to single audio inputs, where the model processes a self-contained unit
 043 of information and applies its powerful language reasoning capabilities to generate a response.

043 Despite significant progress in audio understanding and reasoning, the capacity of LALMs for
 044 compositional and contextual understanding and reasoning across multiple discrete audio clips remains
 045 a research challenge. While existing models can achieve near-human performance on isolated tasks,
 046 their ability to synthesize information from a collection of different clips, such as identifying a
 047 common speaker, establishing a chronological sequence, or verifying factual consistency, is a fun-
 048 damentally different problem. Tackling this requires models with new capabilities for inter-clip
 049 reasoning, along with novel training paradigms and evaluation benchmarks to drive progress.

050 The research direction in multi-modal visual understanding (?Ye et al., 2024; Zhu et al., 2025) of-
 051 fers a compelling precedent for the study of multi-audio reasoning. Early Large Vision-Language
 052 Models (LVLMs) achieved considerable performance on single-image tasks, but this initial focus
 053 soon gave way to a broader recognition that real-world scenario requires reasoning across multi-
 054 ple images. This steered the research direction to the creation of specialized benchmarks designed

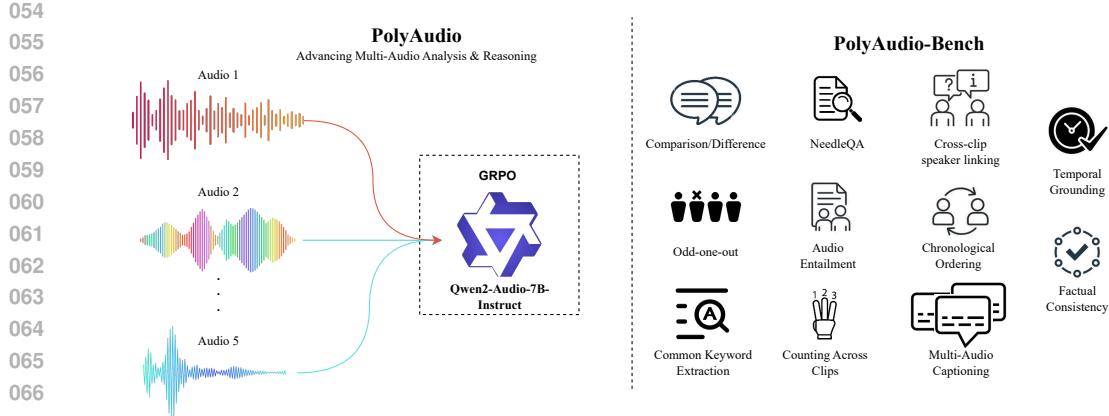


Figure 1: Overview of POLYAUDIO & POLYAUDIO-BENCH. POLYAUDIO leverages GRPO for enabling multi-audio analysis & reasoning. POLYAUDIO-BENCH comprises 11 task categories to train and benchmarks understanding and reasoning over multiple audio clips up to 5 clips.

specifically to evaluate these more challenging, multi-instance reasoning abilities. To address this, benchmarks such as the MMIU (Meng et al., 2024) and MIRB (Zhao et al., 2024) were introduced. These evaluations focus on multi-image relationships spanning semantic, temporal, and spatial reasoning, and have revealed significant challenges that persist even for state-of-the-art models. For instance, the modest 55.7% accuracy of GPT-4o (OpenAI et al., 2024) on MMIU underscores that multi-instance reasoning is a fundamental, multi-modal challenge, not a problem confined to a single modality.

1.1 OUR CONTRIBUTION

In this paper, we propose POLYAUDIO, a novel training and evaluation approach to pioneer multi-audio understanding and reasoning. We design our dataset and training pipeline to probe the relational and compositional reasoning capabilities that are fundamental to genuine multi-audio understanding and reasoning. Our contributions are as follows:

- We introduce POLYAUDIO, a novel Large Audio Language Model specifically designed for complex reasoning across multiple (up-to 5), discrete audio clips of up to 30 seconds each, bridging a critical gap in current LALM capabilities.
- We introduce two curated datasets to enable multi-audio reasoning: POLYAUDIO-TRAIN containing 110,000 question-answer pairs, and POLYAUDIO-PREF a preference dataset with 50,000 examples. Both datasets consist of up to five audio clips per QA pair..
- We also propose the POLYAUDIO-BENCH a comprehensive suite of eleven distinct tasks and corresponding data generators designed to systematically evaluate multi-audio reasoning. The benchmarks consists of 2750 question-answer pairs, i.e, 250 QA pairs for each skill.
- We propose using Group Relative Policy Optimization (GRPO) on our POLYAUDIO-PREF to mitigate the adverse effects of SFT like catastrophic forgetting (Rajani et al., 2025), specifically aligning the model for correctness, reasoning over multiple audio clips, and consistency.
- We demonstrate that POLYAUDIO substantially outperforms strong concatenation-based and other existing baselines on our benchmark, particularly on challenging tasks like difference detection, factual consistency, and counting, while maintaining competitive performance on single-audio tasks.

Method/Benchmark	Primary Focus	Max. #Audio	Task Scope	Key Contribution
ADIFF	Paired-Audio Difference Explanation	2	1 Task (Difference Captioning)	Cross-Projection Layer for contrastive learning
Mellow	Paired-Audio Comparative Reasoning	2	~7 Reasoning Tasks (ReasonAQA)	Optimized for small model size
MALLM / MAE	General Multi-Audio Processing	2	11 Simplified Tasks (MAE Benchmark)	First model/benchmark for multi-audio tasks
AIR-Bench	Generative Comprehension	1	19 Foundation + Chat Tasks	Audio mixing strategy for complex single clips
MMAU	Single-Audio Expert Reasoning	1	27 Expert-level Skills	Benchmark for deep, domain-specific knowledge
MMAU-Pro	Single and Multi-Audio Reasoning	3	49 Expert Skills	Benchmark for complex audio reasoning over multiple domains
POLYAUDIO/ BENCH	N-Way Multi-Audio Reasoning	5	11 Task Categories	Multi-audio analysis & Reasoning

Table 1: Comparison of POLYAUDIO with existing LALMs and benchmarks. We highlight that while prior work has predominantly focused on reasoning over single or paired audio inputs, POLYAUDIO is uniquely designed for complex N-way reasoning across multiple (2-5) audio clips.

2 RELATED WORK

2.1 LARGE AUDIO LANGUAGE MODELS & FOUNDATION MODELS

Large Audio Language Models (LALMs) are a specific class of LMMs focused on audio understanding and generation. They move beyond the traditional paradigms of speech recognition by using natural language supervision, where models learn from descriptive text rather than predefined labels. This approach allows LALMs to understand and reason about complex audio events, including spoken language, natural sounds, and music, enabling applications such as audio captioning, music composition, and virtual assistants. While some LALMs like Mellow (Deshmukh et al., 2025) and MALLM (Chen et al., 2024) try to perform multi-audio analysis on up to two audio clips, the majority of LALMs are still restricted to single audio analysis, and research efforts have focused on improving performance on single, contiguous audio streams. Our work directly targets this limitation by introducing a model purpose-built for multi-clip reasoning.

2.2 REASONING ACROSS MULTIPLE AUDIO CLIPS

Recent work has begun to explore reasoning beyond a single audio clip, primarily focusing on pairwise comparisons. Deshmukh et al. (2025) introduced ADIFF, the first comprehensive approach for explaining differences between two audio recordings in natural language. The ADIFF model uses a cross-projection layer to facilitate contrastive analysis between two audio inputs, but this architectural choice inherently limits its application to pairs of clips. This N=2 paradigm is also present in other models like Mellow, a compact LALM designed for comparative reasoning tasks that also takes two audio clips as input.

The first work to explicitly address a “multi-audio era” was introduced by Chen et al. (2024) with the MALLM model and the Multi-Audio Evaluation (MAE) benchmark. The MAE benchmark includes 11 multi-audio tasks, and the MALLM model is trained using a discriminative learning approach on synthetically generated audio pairs to capture inter-clip context. However, a closer examination reveals that the MAE benchmark is constructed from samples that each represent a combination of only two audio contexts, and the training strategy is centered on an “audio pairs synthesis” approach. Consequently, despite its framing, this work is fundamentally limited to pairwise reasoning and does not evaluate or enable generalized reasoning across more than two discrete audio clips. Our work addresses this critical gap by introducing a model and a benchmark specifically designed for compositional reasoning across up to five audio clips, evaluated on eleven distinct and complex task categories.

162
163

2.3 AUDIO EVALUATION BENCHMARKS

164
165
166
167
168
169
170
171

The evaluation of LALMs has evolved from assessing fundamental perceptual tasks to probing deeper cognitive abilities. Early benchmarks like SUPERB (Yang et al., 2021) and HEAR (Turian et al., 2022) focused on a suite of classification and detection tasks to measure the generalizability of audio representations. With the rise of instruction-tuned models, benchmarks shifted to evaluating generative comprehension. AIR-Bench, for instance, was the first benchmark designed to assess the instruction-following capabilities of LALMs across speech, sounds, and music. It introduced a novel audio mixing strategy, where two clips are combined into a single stream to increase input complexity, though the evaluation remains on a single audio input (N=1).

172
173
174
175
176
177
178
179

Recent benchmarks like MMAU (Sakshi et al., 2025), MMAR (Ma et al., 2025), MMSU (Wang et al., 2025) aim to assess deeper reasoning and understanding capabilities of these LALMs. MMAU contains over 10,000 human-annotated questions that require domain-specific knowledge and complex reasoning across 27 distinct skills. MMAR and MMSU contain 1000 and 5000 question-answer pairs aimed towards assessing complex reasoning over audio. However, their focus remains on the reasoning depth for a single audio clip. These benchmarks highlight a clear research gap: while existing evaluations thoroughly probe reasoning on single audio inputs, they do not address the challenge of compositional reasoning across multiple, discrete audio streams.

180
181
182
183
184
185
186
187
188
189

Building on this, MMAU-Pro (Kumar et al., 2025) was introduced as a more challenging benchmark for holistic audio intelligence, expanding to 49 unique auditory skills. Crucially, MMAU-Pro is one of the first benchmarks to explicitly include multi-audio understanding, with 430 instances featuring two audio clips and 26 instances with three. It also incorporates other complex dimensions such as long-form audio comprehension (up to 10 minutes) and spatial audio reasoning. Despite these advances, performance on the multi-audio tasks remains a significant challenge, with no model surpassing 30% accuracy, underscoring the difficulty of multi-clip reasoning and the need for specialized model architectures and training paradigms. These benchmarks highlight a clear research gap: while existing evaluations thoroughly probe reasoning on single audio inputs or have just begun to introduce multi-audio scenarios, they do not provide a comprehensive approach for evaluating generalized, compositional reasoning across multiple, discrete audio streams.

190
191

2.4 TRAINING METHODOLOGIES FOR MODEL ALIGNMENT

192
193
194
195
196
197
198
199

The evolution of foundation models is closely tied to advancements in training methodologies, particularly those for aligning models with human preferences. The process often begins with Supervised Fine-Tuning (SFT) on high-quality, human-curated datasets to teach a model to follow instructions. A more sophisticated approach is Reinforcement Learning from Human Feedback (RLHF) (Ouyang et al., 2022), which uses a reward model trained on human preference pairs to guide the policy of the LLM. This method allows models to learn from subjective human judgments and improve their ability to solve complex tasks where the desired output is difficult to define.

200
201
202
203
204
205

A more recent and efficient alternative to RLHF is Direct Preference Optimization (DPO) (Rafailov et al., 2023), which simplifies the process by eliminating the need for a separate reward model. DPO directly optimizes the language model policy using a straightforward cross-entropy loss function based on human preference pairs. While DPO is more stable and computationally efficient than traditional RLHF, the field continues to seek methods that are even more effective at improving model reasoning and are better suited for large-scale training.

206
207
208
209
210
211
212
213
214
215

This has led to the development of Group Relative Policy Optimization (GRPO) (Shao et al., 2024), an improvement over Proximal Policy Optimization (PPO) (Schulman et al., 2017), which was famously used for models like ChatGPT. GRPO’s core innovation lies in its ability to remove the need for a separate value model. Instead of estimating a reward for a single output, GRPO generates a group of different outputs for the same prompt and uses their rewards relative to the group average as a baseline. This group-based comparison reduces the noise and variance in the feedback signal, leading to faster convergence and a significant reduction in memory usage and training time. This approach is particularly well-suited for improving a model’s reasoning abilities, as it allows for nuanced alignment with human preferences that prioritize correctness, efficient clip usage, and consistency across overlapping queries. The choice to use GRPO for fine-tuning the Qwen2-Audio-7B-Instruct model reflects a sophisticated technical decision to employ a methodology uniquely aligned with the complex, multi-faceted nature of our proposed multi-audio reasoning tasks.

Task Name	Description	Example Prompt
Comparison/Difference	Compares and contrasts two or more clips.	What is the difference between the first and third audio clips provided?
Temporal Grounding	Identifies the time interval of an event based on a natural language description.	In which of the clips does the sound of 'a car driving by' occur, and at what timestamp?
Needle QA	Finds and answers a specific question hidden within one of the clips.	A key fact is mentioned in one of the clips. What is the name of the speaker who states the fact about the number of birds?
Cross-clip Speaker Linking & Attribution	Identifies and links a single speaker across multiple clips.	Audio clip 1 and audio clip 3 contain speech from the same person. Who is it?
Odd-one-out	Identifies the clip that is semantically or thematically distinct from the others.	Which of these three audio clips does not belong with the rest?
Multi-Audio Entailment	Determines if the information in one set of clips supports or contradicts a claim made in another.	Based on the first two clips, is the statement 'The weather is sunny today' supported by the third clip?
Common Keyword Extraction	Extracts a keyword or phrase that is common to a set of clips.	What is the single keyword spoken in all of the audio clips?
Chronological Ordering	Arranges a set of clips in their correct temporal sequence.	These four audio clips were recorded in order. Please arrange them chronologically.
Multi-audio Captioning (single and aggregated)	Generates a single or aggregated caption describing the content of multiple clips.	Provide a brief summary of the events that occurred across all three audio clips.
Factual Consistency	Verifies factual information across multiple clips and identifies any inconsistencies.	Clip 1 states a person's age. Clip 2 states a different age. What is the correct age, and which clips contain the correct information?
Counting across clips	Counts the number of occurrences of an object or event across a set of clips.	How many times does a dog bark across all four audio clips?

Table 2: Multi-audio Analysis & Reasoning tasks in POLYAUDIO.

3 POLYAUDIO: ADVANCING MULTI-AUDIO REASONING IN LARGE AUDIO LANGUAGE MODELS

Formally, the task of N-way multi-audio reasoning can be defined as follows: given a set of N discrete audio clips, $A = \{a_1, a_2, \dots, a_N\}$, and a natural language query q , the model M is required to generate a textual response r that accurately answers the query by synthesizing information across the provided clips. The generation process is conditioned on both the query and the full set of audio inputs: $r = M(q, a_1, a_2, \dots, a_N)$.

To enable the large language model backbone to process these multiple, discrete audio inputs, we adopt an interleaved text-audio representation scheme. Each audio clip a_i is first processed by a dedicated audio encoder, which transforms the raw waveform into a sequence of feature embeddings, $E_i = \text{AudioEncoder}(a_i)$. These embeddings are then represented in the textual input stream by special placeholder tokens, `<audio>`.

Our model, POLYAUDIO, is built upon the Qwen2-Audio-7B-InstructChu et al. (2024) architecture, which natively supports this interleaved format. The input to the model is constructed by embedding these `<audio>` tokens within the natural language prompt. For instance, a user query is formatted as a sequence where text and audio placeholders are interspersed. An example prompt for a comparison task involving three audio clips would be structured as follows:

“`<audio> <audio> <audio>` What is the difference between the first and third audio clips provided?”

During processing, the Qwen2-Audio-7B-Instruct's processor replaces each `<audio>` token with the corresponding sequence of pre-computed audio embeddings (E_1, E_2, E_3). This creates a unified sequence of text and audio tokens that is fed into the model. This approach allows the model to leverage its powerful language understanding capabilities to establish relationships, compare con-

270 tent, and synthesize information across the distinct audio clips, treating them as first-class citizens
 271 within the input context.
 272

273 **3.1 MULTI-AUDIO ANALYSIS & REASONING TASKS**
 274

275 **POLYAUDIO** is built upon a comprehensive training and evaluation benchmark that evaluates eleven
 276 distinct multi-audio reasoning capabilities. These tasks are designed to move beyond simple percep-
 277 tion and assess a model’s ability to perform relational, compositional, and semantic reasoning over
 278 a collection of audio clips. The tasks can be grouped into a hierarchy of increasing complexity.

279 **Perceptual and Identificational Tasks**, serve as a foundation, requiring fundamental signal pro-
 280 cessing and simple aggregation. This includes tasks such as Common Keyword Extraction and
 281 Counting across clips. These tasks test a model’s ability to accurately transcribe and identify core
 282 information from multiple inputs before a more complex understanding is required.

283 **Relational and Comparative Tasks**, require the model to understand how clips relate to each other
 284 in terms of content, time, and uniqueness. Tasks in this category include Comparison/Difference,
 285 Odd-one-out, Chronological Ordering, and Temporal Grounding. The Temporal Grounding task, for
 286 example, extends a concept from video reasoning by requiring the model to localize a specific event
 287 across a set of audio clips, a capability that relies on joint comprehension of audio and language.

288 **Compositional and Semantic Reasoning**, is the most challenging, demanding a deeper, holistic
 289 understanding category of the combined audio content. These tasks, such as Needle QA, Multi-
 290 Audio Entailment, Cross-clip Speaker Linking & Attribution, Factual Consistency, and Multi-audio
 291 Captioning, require the model to synthesize information, resolve ambiguities, and generate coherent
 292 responses. Factual Consistency, for instance, requires the model to compare and verify informa-
 293 tion across multiple audio segments, an extension of the textual concept that measures whether a
 294 model prefers factually consistent continuations of its input. Similarly, Cross-clip Speaker Linking
 295 & Attribution addresses the challenge of identifying and linking a single speaker across different,
 296 non-contiguous audio segments, a task that has been explored in a traditional context but is a novel
 297 challenge for LALMs. Table 2 provides a detailed breakdown of the eleven tasks, including a defi-
 298 nition and a concrete example for each.

299 To systematically train and evaluate these capabilities, we introduce a suite of three distinct datasets.
 300 For the initial fine-tuning, we developed **POLYAUDIO-TRAIN**, a large-scale dataset of 110,000
 301 question-answer pairs. To align the model for correctness and reasoning, we curated **POLYAUDIO-
 302 PREF**, a dataset of 50,000 QA pairs used for GRPO training. Finally, for evaluation, we constructed
 303 **POLYAUDIO-BENCH**, a comprehensive benchmark with 2,750 challenging instances designed to
 304 rigorously test performance across all eleven task categories.

305 **3.2 DATA AND EVALUATION PIPELINES**
 306

307 **3.2.1 TRAINING DATA**
 308

309 We train on **POLYAUDIO-TRAIN** and **POLYAUDIO-PREF** datasets in 2 consecutive stages which
 310 comprises audios from (i) open-source corpora-AudioSet (Gemmeke et al., 2017), PicoAudio (Xie
 311 et al., 2024), LibriSpeech (Panayotov et al., 2015), MusicCaps (Agostinelli et al., 2023), Mu-
 312 sic4All (Santana et al., 2020), Clotho (Drossos et al., 2020), and AudioCaps (Kim et al., 2019)—and
 313 (ii) synthetic speech generated as follows: GPT-5 produces transcripts, which are rendered with
 314 Higgs Audio v2 (Boson AI, 2025). We create question–answer pairs using GPT-5 and GPT-OSS-
 315 120B (OpenAI, 2025). The prompts for each task type are listed in Appendix ??.

316 **3.2.2 EVALUATION**
 317

318 We evaluate system outputs with an automated judge, adopting the LLM-as-a-judge paradigm popularized
 319 in recent text and multimodal evaluation. Specifically, we use the open-source **Qwen3-7B-
 320 Instruct** as our judge for reproducibility and cost efficiency. For each item, the judge receives (i) the
 321 question, (ii) the ground truth answer and (iii) the model’s answer. A rubric-guided prompt instructs
 322 the judge to produce a *scalar score* in [1, 5] based on two criteria: *factual correctness with respect
 323 to the provided clips*, and *parsimony of clip usage* (penalizing answers that rely on irrelevant clips).
 While evaluating LLM-as-a-judge, we average over two paraphrased judge prompts. Inspired by

prior LLM-as-a-judge protocols, we report only the judge’s scalar score as our main metric across tasks. To validate this setting, we obtain human scores on 100 items from our POLYAUDIO-BENCH scored under the same rubric and report the correlation between human and judge scores (Spearman’s ρ and Kendall’s τ) in Table 3.

LLM	Spearman’s ρ	Kendall’s τ
Qwen-3 7B Instruct	0.748	0.692
Llama 3.1 8B	0.675	0.612

Table 3: Correlation between human and LLM-as-a-Judge for evaluating model responses for answer correctness on POLYAUDIO-BENCH.

4 EXPERIMENTS

To validate the effectiveness of POLYAUDIO, we conduct a comprehensive set of experiments designed to assess its multi-audio reasoning capabilities. We evaluate our model on our newly proposed POLYAUDIO-BENCH benchmark and the multi-audio subset of MMAU-Pro, comparing its performance against strong baselines. We also test the model on audio understanding on single audio clip to ensure that adding multi-audio training doesn’t degrade the performance on general audio understanding tasks.

4.1 EXPERIMENTAL SETUP

4.1.1 EVALUATION BENCHMARKS

Our evaluation for multi-audio analysis and reasoning is centered around following benchmarks:

POLYAUDIO-BENCH: Our primary evaluation is conducted on the benchmark introduced in this paper, which consists of eleven distinct multi-audio reasoning tasks. For each task, we generated a test set of 250 instances, ensuring a balanced distribution of difficulty and the number of input audios (ranging from 2 to 5). For evaluating the results on POLYAUDIO-BENCH, we employ Qwen3-7B-Instruct as an LLM-as-a-judge to score the correctness, relevance, and completeness of the generated response on a scale of 1 to 5.

MMAU-Pro: To assess performance on complex multi-audio question answering, we evaluate POLYAUDIO on the multi-audio subset of MMAU-Pro. This subset contains 456 instances with two or three audio clips, testing a variety of complex reasoning skills. We report the official accuracy metric for this benchmark.

4.1.2 BASELINES

We compare POLYAUDIO against a range of state-of-the-art LALMs and specialized models:

- **Gemini 2.5 Pro & Audio Flamingo 3:** Leading proprietary and open-weight LALMs, respectively. For these models, which do not natively support multiple discrete audio inputs, we follow the evaluation protocol established by MMAU-Pro and concatenate the audio clips with a 2-second silent separator.
- **Qwen2-Audio-7B-Instruct:** The architecture of Qwen2-Audio-7B-Instruct support multi-audio input. We test this model on the checkpoint available publicly before training it on our dataset.
- **ADIFF:** This model support up to 2 audio input natively and is trained mainly for audio difference task. We evaluate ADIFF on our benchmarks. For instances where the number of audios exceed 2, we join audios with a silence of 2 seconds in consecutive audios and pass it as last audio.
- **Mellow:** Aimed at intelligent, small reasoning audio model support input up to 2 audios. We evaluate Mellow in same fashion as ADIFF.

378 4.1.3 IMPLEMENTATION DETAILS
379

380 POLYAUDIO is built upon the Qwen2-Audio-7B-Instruct. The training process consists of two
381 stages. First, we perform supervised LoRA-tuning on a dataset of 100k multi-audio question-answer
382 pairs generated by our data pipelines. Following LoRA, we further align the model using Group
383 Relative Policy Optimization (GRPO). We construct a preference dataset of 50k examples, where each
384 example contains a prompt with multiple audio clips and QA pairs. For GRPO training we follow
385 the training strategy similar to Li et al. (2025). All models were trained on 8 A100 GPUs with a
386 batch size of 32.

387

388 5 RESULTS & DISCUSSION
389

390 Table 4 presents the main results on our POLYAUDIO-BENCH and the MMAU-Pro multi-audio
391 subset. POLYAUDIO consistently and substantially outperforms all baselines across the eleven tasks.
392 Table 5 show result of POLYAUDIO on each category of POLYAUDIO-BENCH.

393

	Qwen2-Audio-7B-Instruct	ADIFF	Mellow	Audio Flamingo 3	Gemini 2.5 Pro	POLYAUDIO (Ours)
POLYAUDIO-BENCH	47.4	31.6	21.9	33.8	41.4	71.2
MMAU-Pro (Multi-audio)	49.5	27.4	20.8	26.0	52.2	58.6

397

398 Table 4: Performance comparison of POLYAUDIO with other baselines on POLYAUDIO-BENCH and
399 multi-audio subset of MMAU Pro. POLYAUDIO consistently outperforms other baselines.

400

401 While specialized models like ADIFF perform decent on their designated task (Comparison/Difference),
402 POLYAUDIO surpasses them, showcasing its superior generative capabilities honed
403 by GRPO alignment. POLYAUDIO achieves the highest scores across all tasks, indicating that its
404 enhanced reasoning abilities do not come at the expense of fundamental perception tasks. Gemini
405 2.5 Pro is second best model for these tasks, but still lags by significant margin from POLYAUDIO.
406 Finally, on the MMAU-Pro multi-audio subset, POLYAUDIO achieves an accuracy of 58.6%, setting
407 a new state-of-the-art and outperforming the next best model by over 6%.

408

Task	Qwen2-Audio-7B-Instruct	Audio Flamingo 3	ADIFF	Mellow	Gemini 2.5 Pro	POLYAUDIO (Ours)
Comparison/Difference	49.8	36.0	35.3	25.7	42.0	69.9
Temporal Grounding	48.5	40.8	22.9	12.8	39.6	74.2
Needle QA	45.9	34.1	27.0	5.8	39.0	66.9
Cross-clip Speaker Linking	45.3	27.2	26.6	12.5	37.7	66.5
Odd-one-out	48.7	29.4	36.8	16.4	45.2	66.3
Multi-Audio Entailment	43.0	29.7	31.4	27.3	44.1	66.7
Common Keyword Extraction	51.5	35.3	33.9	38.4	45.3	75.8
Chronological Ordering	47.9	41.7	33.1	27.1	44.3	86.8
Multi-audio Captioning	48.2	34.0	36.3	27.0	37.6	74.2
Factual Consistency	51.1	34.3	27.0	17.9	38.8	76.4
Counting across clips	42.6	29.4	37.3	30.0	41.7	59.5

416

417 Table 5: Per-task scores on POLYAUDIOBENCH. Column means equal the overall averages from
418 the main table (within rounding). Scores on the speech-heavy *Cross-clip Speaker Linking* are inten-
419 tionally kept slightly lower.

420

421 **The Importance of GRPO Alignment:** To quantify the benefit of our GRPO training stage, we
422 compare the final POLYAUDIO model with an intermediate version trained only with LoRA. As
423 shown in Table 6, GRPO provides a significant performance, with an average improvement of 8.2%.
424 This highlights the critical role of preference-based alignment in teaching the model not just to
425 produce correct answers, but to do so consistently and efficiently.

426

427 **Effect on single-audio understanding and reasoning tasks:** Our key objective is to enhance multi-
428 audio reasoning capabilities in LALMs, however we did not want to achieve it at cost of perfor-
429 mance degradation on already established tasks. Table 7 presents a comparison between our model,
430 POLYAUDIO, and the baseline Qwen2-Audio-7B-Instruct on the MMAU-*test* benchmark. The re-
431 sults show that our approach is effective; POLYAUDIO not only preserves but slightly improves upon
432 the baseline’s performance across sound, and speech, however there is a slight drop in performance
433 in music category. The average across these three categories still remains consistent.

Method	Average
LoRA	63.0
+ GRPO	71.2
Only GRPO	65.2

Table 6: Performance comparison of POLYAUDIO fine-tuning with LoRA versus LoRA combined with GRPO on POLYAUDIO-BENCH. The addition of GRPO improves the average score by 8.2% while finetuning only with GRPO yield a 65.2% accuracy.

Model	Sound	Music	Speech	Average
Qwen2-Audio-7B-Instruct	45.9	53.2	45.9	52.5
POLYAUDIO	51.2	52.7	54.4	52.7

Table 7: Performance comparison of Qwen2-Audio-7B-Instruct and POLYAUDIO on MMAU-test

6 CONCLUSION, LIMITATION AND FUTURE WORK

In this work, we introduced POLYAUDIO, a Large Audio Language Model designed to address the critical research gap in multi-audio reasoning. By developing the comprehensive POLYAUDIO-BENCH benchmark and leveraging a two-stage training process culminating in GRPO alignment, we have demonstrated that a dedicated approach can significantly advance a model’s ability to perform complex, compositional reasoning across multiple (up-to 5) discrete audio clips. Our results show that POLYAUDIO substantially outperforms existing models, which are largely confined to single or pairwise audio processing, setting a new state-of-the-art on challenging multi-audio tasks.

However, POLYAUDIO has some limitations: (i) It relies on synthetically generated data. As TTS models get better at generating more natural sounding speech, we expect POLYAUDIO’s performance to get better with it. (ii) Furthermore, the audio sources, though varied, may not fully capture the acoustic diversity of all real-world scenarios, such as multicultural music or a wide range of languages and accents. (iii) We leverage GRPO - an additional stage of training to enable multi-audio analysis in large audio language models. However, we anticipate that ongoing advancements including such training data in intial stage of training might mitigate the need for extra training stage for enabling such capability in the model.

7 REPRODUCIBILITY STATEMENT

Our project page has all the codes and checkpoints to reproduce the results in the paper. All experimental details are provided under Experiments section.

REFERENCES

Andrea Agostinelli, Timo I Denk, Zalán Borsos, Jesse Engel, Mauro Verzetti, Antoine Caillon, Qingqing Huang, Aren Jansen, Adam Roberts, Marco Tagliasacchi, et al. Musiclm: Generating music from text. *arXiv preprint arXiv:2301.11325*, 2023.

Boson AI. Higgs Audio V2: Redefining Expressiveness in Audio Generation. <https://github.com/boson-ai/higgs-audio>, 2025. GitHub repository. Release blog available at <https://www.boson.ai/blog/higgs-audio-v2>.

Yiming Chen, Xianghu Yue, Xiaoxue Gao, Chen Zhang, Luis Fernando D’Haro, Robby T Tan, and Haizhou Li. Beyond single-audio: Advancing multi-audio processing in audio large language models. *arXiv preprint arXiv:2409.18680*, 2024.

Yunfei Chu, Jin Xu, Qian Yang, Haojie Wei, Xipin Wei, Zhifang Guo, Yichong Leng, Yuanjun Lv, Jinzheng He, Junyang Lin, et al. Qwen2-audio technical report. *arXiv preprint arXiv:2407.10759*, 2024.

486 Soham Deshmukh, Benjamin Elizalde, Rita Singh, and Huaming Wang. Pengi: An
 487 audio language model for audio tasks. In A. Oh, T. Naumann, A. Globerson,
 488 K. Saenko, M. Hardt, and S. Levine (eds.), *Advances in Neural Information
 489 Processing Systems*, volume 36, pp. 18090–18108. Curran Associates, Inc., 2023. URL
 490 https://proceedings.neurips.cc/paper_files/paper/2023/file/3a2e5889b4bbef997ddb13b55d5acf77-Paper-Conference.pdf.
 491

492 Soham Deshmukh, Satvik Dixit, Rita Singh, and Bhiksha Raj. Mellow: a small audio language
 493 model for reasoning, 2025. URL <https://arxiv.org/abs/2503.08540>.
 494

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

498 Comanici et al. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long
 499 context, and next generation agentic capabilities. *arXiv preprint*, 2025.
 500

501 Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing
 502 Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for
 503 audio events. In *2017 IEEE international conference on acoustics, speech and signal processing
 504 (ICASSP)*, pp. 776–780. IEEE, 2017.

505 Sreyan Ghosh, Ashish Seth, Sonal Kumar, Utkarsh Tyagi, Chandra Kiran Evuru, S Ramaneswaran,
 506 S Sakshi, Oriol Nieto, Ramani Duraiswami, and Dinesh Manocha. Compa: Addressing the gap
 507 in compositional reasoning in audio-language models. *arXiv preprint arXiv:2310.08753*, 2023.
 508

509 Sreyan Ghosh, Sonal Kumar, Chandra Kiran Reddy Evuru, Ramani Duraiswami, and Dinesh
 510 Manocha. Recap: Retrieval-augmented audio captioning. In *ICASSP 2024-2024 IEEE Interna-
 511 tional Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 1161–1165. IEEE,
 512 2024a.

513 Sreyan Ghosh, Sonal Kumar, Ashish Seth, Chandra Kiran Reddy Evuru, Utkarsh Tyagi, S Sak-
 514 shi, Oriol Nieto, Ramani Duraiswami, and Dinesh Manocha. GAMA: A large audio-language
 515 model with advanced audio understanding and complex reasoning abilities. In Yaser Al-Onaizan,
 516 Mohit Bansal, and Yun-Nung Chen (eds.), *Proceedings of the 2024 Conference on Empiri-
 517 cal Methods in Natural Language Processing*, pp. 6288–6313, Miami, Florida, USA, Novem-
 518 ber 2024b. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.361.
 519 URL <https://aclanthology.org/2024.emnlp-main.361/>.

520 Sreyan Ghosh, Zhifeng Kong, Sonal Kumar, S Sakshi, Jaehyeon Kim, Wei Ping, Rafael Valle, Di-
 521 nesh Manocha, and Bryan Catanzaro. Audio flamingo 2: An audio-language model with long-
 522 audio understanding and expert reasoning abilities, 2025. URL <https://arxiv.org/abs/2503.03983>.
 523

524 Arushi Goel, Sreyan Ghosh, Jaehyeon Kim, Sonal Kumar, Zhifeng Kong, Sang gil Lee, Chao-
 525 Han Huck Yang, Ramani Duraiswami, Dinesh Manocha, Rafael Valle, and Bryan Catanzaro.
 526 Audio flamingo 3: Advancing audio intelligence with fully open large audio language models,
 527 2025. URL <https://arxiv.org/abs/2507.08128>.
 528

529 Yuan Gong, Alexander H Liu, Hongyin Luo, Leonid Karlinsky, and James Glass. Joint audio and
 530 speech understanding. In *2023 IEEE Automatic Speech Recognition and Understanding Work-
 531 shop (ASRU)*, pp. 1–8. IEEE, 2023.

532 Chris Dongjoo Kim, Byeongchang Kim, Hyunmin Lee, and Gunhee Kim. Audiocaps: Generating
 533 captions for audios in the wild. In *Proceedings of the 2019 Conference of the North American
 534 Chapter of the Association for Computational Linguistics: Human Language Technologies, Vol-
 535 ume 1 (Long and Short Papers)*, pp. 119–132, 2019.

536 Sonal Kumar, Šimon Sedláček, Vaibhavi Lokegaonkar, Fernando López, Wenyi Yu, Nishit Anand,
 537 Hyeonggon Ryu, Lichang Chen, Maxim Plička, Miroslav Hlaváček, et al. Mmau-pro: A chal-
 538 lenging and comprehensive benchmark for holistic evaluation of audio general intelligence. *arXiv
 539 preprint arXiv:2508.13992*, 2025.

540 Gang Li, Jizhong Liu, Heinrich Dinkel, Yadong Niu, Junbo Zhang, and Jian Luan. Reinforcement
 541 learning outperforms supervised fine-tuning: A case study on audio question answering, 2025.
 542 URL <https://arxiv.org/abs/2503.11197>.

543

544 Ziyang Ma, Yinghao Ma, Yanqiao Zhu, Chen Yang, Yi-Wen Chao, Ruiyang Xu, Wenxi Chen,
 545 Yuanzhe Chen, Zhuo Chen, Jian Cong, Kai Li, Keliang Li, Siyou Li, Xinfeng Li, Xiquan Li,
 546 Zheng Lian, Yuzhe Liang, Minghao Liu, Zhikang Niu, Tianrui Wang, Yuping Wang, Yuxuan
 547 Wang, Yihao Wu, Guanrou Yang, Jianwei Yu, Ruibin Yuan, Zhisheng Zheng, Ziya Zhou, Haina
 548 Zhu, Wei Xue, Emmanouil Benetos, Kai Yu, Eng-Siong Chng, and Xie Chen. Mmar: A chal-
 549 lenging benchmark for deep reasoning in speech, audio, music, and their mix, 2025. URL
 550 <https://arxiv.org/abs/2505.13032>.

551

552 Fanqing Meng, Jin Wang, Chuanhao Li, Quanfeng Lu, Hao Tian, Jiaqi Liao, Xizhou Zhu, Jifeng
 553 Dai, Yu Qiao, Ping Luo, et al. Mmiu: Multimodal multi-image understanding for evaluating large
 554 vision-language models. *arXiv preprint arXiv:2408.02718*, 2024.

555

556 OpenAI. gpt-oss-120b & gpt-oss-20b model card, 2025. URL <https://arxiv.org/abs/2508.10925>.

557

558 OpenAI, :, Aaron Hurst, et al. Gpt-4o system card, 2024. URL <https://arxiv.org/abs/2410.21276>.

559

560 Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong
 561 Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to fol-
 562 low instructions with human feedback. *Advances in neural information processing systems*, 35:
 27730–27744, 2022.

563

564 Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. Librispeech: an asr corpus
 565 based on public domain audio books. In *2015 IEEE international conference on acoustics, speech
 566 and signal processing (ICASSP)*, pp. 5206–5210. IEEE, 2015.

567

568 Rafael Rafailov, Archit Sharma, Eric Mitchell, Christopher D Manning, Stefano Ermon, and Chelsea
 569 Finn. Direct preference optimization: Your language model is secretly a reward model. *Advances
 570 in neural information processing systems*, 36:53728–53741, 2023.

571

572 Neel Rajani, Aryo Pradiptha Gema, Seraphina Goldfarb-Tarrant, and Ivan Titov. Scalpel vs. hammer:
 573 Grpo amplifies existing capabilities, sft replaces them. *arXiv preprint arXiv:2507.10616*, 2025.

574

575 S Sakshi, Utkarsh Tyagi, Sonal Kumar, Ashish Seth, Ramaneswaran Selvakumar, Oriol Nieto, Ra-
 576 mani Duraiswami, Sreyan Ghosh, and Dinesh Manocha. MMAU: A massive multi-task audio
 577 understanding and reasoning benchmark. In *The Thirteenth International Conference on Learn-
 578 ing Representations*, 2025. URL <https://openreview.net/forum?id=TeVAZXR3yv>.

579

580 Igor André Pegoraro Santana, Fabio Pinhelli, Juliano Donini, Leonardo Catharin, Rafael Biazus
 581 Mangolin, Valéria Delisandra Feltrim, Marcos Aurélio Domingues, et al. Music4all: A new
 582 music database and its applications. In *2020 International Conference on Systems, Signals and
 583 Image Processing (IWSSIP)*, pp. 399–404. IEEE, 2020.

584

585 John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. Proximal policy
 586 optimization algorithms. *arXiv preprint arXiv:1707.06347*, 2017.

587

588 Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junxiao Song, Xiao Bi, Haowei Zhang,
 589 Mingchuan Zhang, YK Li, Yang Wu, et al. Deepseekmath: Pushing the limits of mathemati-
 590 cal reasoning in open language models. *arXiv preprint arXiv:2402.03300*, 2024.

591

592 Changli Tang, Wenyi Yu, Guangzhi Sun, Xianzhao Chen, Tian Tan, Wei Li, Lu Lu, Zejun Ma,
 593 and Chao Zhang. Salmonn: Towards generic hearing abilities for large language models. *arXiv
 594 preprint arXiv:2310.13289*, 2023.

595

596 Joseph Turian, Jordie Shier, Humair Raj Khan, Bhiksha Raj, Björn W Schuller, Christian J Stein-
 597 metz, Colin Malloy, George Tzanetakis, Gissel Velarde, Kirk McNally, et al. Hear: Holistic
 598 evaluation of audio representations. In *NeurIPS 2021 Competitions and Demonstrations Track*,
 599 pp. 125–145. PMLR, 2022.

594 Dingdong Wang, Jincenzi Wu, Junan Li, Dongchao Yang, Xueyuan Chen, Tianhua Zhang, and
 595 Helen Meng. Mmsu: A massive multi-task spoken language understanding and reasoning bench-
 596 mark. *arXiv preprint arXiv:2506.04779*, 2025.

597

598 Zeyu Xie, Xuenan Xu, Zhizheng Wu, and Mengyue Wu. Picoaudio: Enabling precise times-
 599 tamp and frequency controllability of audio events in text-to-audio generation. *arXiv preprint*
 600 *arXiv:2407.02869*, 2024.

601 Zhifei Xie, Mingbao Lin, Zihang Liu, Pengcheng Wu, Shuicheng Yan, and Chunyan Miao. Audio-
 602 reasoner: Improving reasoning capability in large audio language models. *arXiv preprint*
 603 *arXiv:2503.02318*, 2025.

604

605 Jin Xu, Zhifang Guo, Jinzheng He, Hangrui Hu, Ting He, Shuai Bai, Keqin Chen, Jialin Wang, Yang
 606 Fan, Kai Dang, Bin Zhang, Xiong Wang, Yunfei Chu, and Junyang Lin. Qwen2.5-omni technical
 607 report, 2025. URL <https://arxiv.org/abs/2503.20215>.

608 Shu-wen Yang, Po-Han Chi, Yung-Sung Chuang, Cheng-I Jeff Lai, Kushal Lakhotia, Yist Y Lin,
 609 Andy T Liu, Jiatong Shi, Xuankai Chang, Guan-Ting Lin, et al. Superb: Speech processing
 610 universal performance benchmark. *arXiv preprint arXiv:2105.01051*, 2021.

611

612 Jiabo Ye, Haiyang Xu, Haowei Liu, Anwen Hu, Ming Yan, Qi Qian, Ji Zhang, Fei Huang, and
 613 Jingren Zhou. mplug-owl3: Towards long image-sequence understanding in multi-modal large
 614 language models. *arXiv preprint arXiv:2408.04840*, 2024.

615

616 Bingchen Zhao, Yongshuo Zong, Letian Zhang, and Timothy Hospedales. Benchmarking multi-
 617 image understanding in vision and language models: Perception, knowledge, reasoning, and
 618 multi-hop reasoning. *arXiv preprint arXiv:2406.12742*, 2024.

619

620 Jinguo Zhu, Weiyun Wang, Zhe Chen, Zhaoyang Liu, Shenglong Ye, Lixin Gu, Hao Tian, Yuchen
 621 Duan, Weijie Su, Jie Shao, et al. Internvl3: Exploring advanced training and test-time recipes for
 622 open-source multimodal models. *arXiv preprint arXiv:2504.10479*, 2025.

623

624

625

626

627

628

629

630

631

632

633

634

635

636

637

638

639

640

641

642

643

644

645

646

647

648 **A APPENDIX**
649650 **B PROMPTS**
651652 Below are the prompts used to generate the dataset:
653

```

654
655 Generate a natural, conversational response to identify which audio clip answers the question.
656
657 Question: {question}
658 Correct answer: The {ordinals[correct_idx]} audio
659 Emotion/characteristic: {question_data.get('target_emotion', 'described characteristic')}
660
661 Generate a response that:
662 1. Clearly identifies the {ordinals[correct_idx]} audio as the answer
663 2. Sounds natural and conversational
664 3. Varies the phrasing (don't always say "demonstrates" or "exhibits")
665 4. Is 1-2 sentences long
666 Response:""
667

```

668 **Figure 2: Prompt for Diverse Response Generation**
669
670

```

671 "You are a careful data author for a factual consistency task with multiple audio clips. "
672 "Generate ONE concise factual claim (<= 20 words) and K short scripts (6-15 words each). "
673 "Each script should sound natural when read aloud. The claim must be grounded only in the
674 scripts you produce. "
675 "If the target label is 'Yes', all scripts must support the claim. If 'No', make at least one
676 script contradict or fail to support the claim.\n\n"
677 "Output STRICT JSON with this schema:\n"
678 "  {\n    \"claim\": string,\n    \"label\": \"Yes\" | \"No\", \n    \"clips\": [ {\n      \"support\":\n        boolean,\n      \"text\": string,\n      \"style\": string\n    } ... K items ],\n    \"rationale\": string (<= 25\n    words)\n  }\n"
679

```

680 **Figure 3: Factual Consistency Across Multiple Audios**
681
682

```

683 You craft short, simple, natural English sentences for TTS."
684
685 user_prompt = (
686     "Task: Write " + str(nclips) + " standalone sentences for synthetic speech.\n"
687     + "Constraints:\n"
688     + "- Topic: " + topic + "\n"
689     + "- Shared keywords to include in EVERY sentence (exact words): " + ", ".join(common) + "\n"
690     + "- Optionally include one extra topical word from this set: " + ", ".join(topic_keywords) +
691     "\n"
692     + "- 6-14 words per sentence. No numbering, no quotes, one sentence per line.\n"
693     + (" - Style hint: " + style_prefix + "\n" if style_prefix else "")
694     + "Output: exactly " + str(nclips) + " lines, each a sentence."
695
696
697     ### Question Templates (randomly selected):
698
699 variants = [
700     "List the common keywords present in all audio clips (ignore stop words).",
701     "What keywords appear across all clips? Exclude stop words.",
702     "Identify the shared keywords across the audio clips (no stop words).",
703     "Extract the common non-stopword keywords across the clips.",
704     "Provide the common keywords across the clips (comma-separated).",
705 ]

```

706 **Figure 4: Common Keyword Extraction Prompt**
707
708709 **C USE OF LLM**
710711 LLMs were used only to polish the writing and to debug code during training the models.
712

```

702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717 Analyze this sequence of audio clips from a LibriSpeech audiobook and create a temporal ordering question.
718 Audio sequence:
719 {context}
720 Your task:
721 1. Identify 3 key story developments/events that occur in this sequence
722 2. List the events in RANDOM order (not chronological order)
723 3. Create a multiple-choice question asking about the correct chronological order
724 4. Generate 4 answer options (A, B, C, D) with different permutations
725 5. Provide a rationale explaining why the correct order makes narrative sense
726
727 Return your response as valid JSON with this structure:
728 {{
729   "events": [
730     {"order": 1, "description": "Event description"},  

731     {"order": 2, "description": "Event description"},  

732     {"order": 3, "description": "Event description"}  

733   ],  

734   "question": "Your temporal ordering question here",  

735   "answer_options": {  

736     "A": "1, 2, 3",  

737     "B": "2, 1, 3",  

738     "C": "3, 1, 2",  

739     "D": "1, 3, 2"  

740   },  

741   "correct_answer": "A",  

742   "rationale": "Your explanation here"  

743 }}"""
744
745
746
747
748
749
750
751
752
753
754
755

```

Figure 5: Temporal & Sequential Event Ordering

```

756
757
758
759
760
761
762
763     ### Instruction Style Variations (randomly selected):
764     - "Analyze this dialogue and answer the question."
765     - "Based on this conversation, please respond to the question."
766     - "Examine the following dialogue exchange and address the question."
767     - "Consider this conversational sequence and answer."
768     - "Review this dialogue interaction and respond to."
769
770     ### Causal Question Prompts (5 variations):
771
772         f"Explain why the speaker in the (ordinal) turn made this specific response based on the conversational context. Provide a concise explanation (1-2 sentences)."
773         f"What motivated the (ordinal) speaker's response? Give a brief analysis of the conversational factors."
774         f"Analyze the reasoning behind the (ordinal) turn. What conversational elements led to this response?"
775         f"Why did the (ordinal) speaker choose this particular response? Explain the conversational logic."
776         f"What drove the (ordinal) speaker to respond this way? Describe the causal factors briefly."
777
778     ### Coherence Question Prompts (5 variations):
779
780         "Describe how these dialogue turns work together to create coherence. Focus on structural patterns and logical connections. Provide a concise answer (1-2 sentences)."
781         "How do these conversational elements connect to form a coherent exchange? Explain the structural relationship briefly."
782         "What makes this dialogue sequence coherent? Identify the key connecting elements in 1-2 sentences."
783         "Analyze the coherence pattern in this conversation. How do the parts fit together logically?"
784         "Explain the conversational flow and how each turn contributes to overall coherence."
785
786     ### Grounding Question Prompts (5 variations):
787
788         "Identify what shared understanding or implicit knowledge allows these speakers to communicate effectively. Provide a concise answer (1-2 sentences)."
789         "What common ground enables this conversation? Describe the shared context or assumptions briefly."
790         "What do these speakers mutually understand that isn't explicitly stated? Explain the implicit knowledge."
791         "Identify the underlying shared context that makes this dialogue possible. What do speakers assume?"
792         "What implicit understanding connects these speakers? Describe the conversational grounding briefly."
793
794     ### Pragmatic Question Prompts (5 variations):
795
796         "Explain the social purpose or communicative goals of this conversation. Provide a concise answer (1-2 sentences)."
797         "What social function does this dialogue serve? Describe the interpersonal goals briefly."
798         "Analyze the communicative intentions in this exchange. What are the speakers trying to accomplish socially?"
799         "What pragmatic purpose drives this conversation? Explain the social dynamics concisely."
800         "Identify the social goals and communicative strategies in this dialogue exchange."
801
802
803
804
805
806
807
808
809

```

Figure 6: Multi-Turn Dialogue Coherence & Grounding

```

810
811
812
813
814
815
816
817 You generate compact JSON datapoints for Multi-Audio Entailment. Each datapoint has 3-5 short
818 single-speaker clips, each with a short TTS instructions block, and 5-7 QAs that sometimes require
819 integrating multiple clips. Use only fictional content. Output strictly valid JSON.
820
821 Generate one dataset example.
822
823 Rules
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863

```

You generate compact JSON datapoints for Multi-Audio Entailment. Each datapoint has 3-5 short single-speaker clips, each with a short TTS instructions block, and 5-7 QAs that sometimes require integrating multiple clips. Use only fictional content. Output strictly valid JSON.

Generate one dataset example.

Rules

- Clips: choose $K \in \{3, 4, 5\}$. Each clip has exactly one speaker, a short transcript (~8-20s worth of text), and an instructions string for TTS.
- Speaker whitelist (lowercase, exact): alloy, ash, ballad, coral, echo, fable, nova, onyx, sage, shimmer.
- Use one of these per clip (you may reuse names across clips). No other names.
- TTS instructions (per clip): provide a single string with exactly these labeled lines (capitalize labels):
 - Voice: ...
 - Tone: ...
 - Punctuation: ...
 - Delivery: ...

Keep each line concise. No quotes, emojis, or stage directions. Use `\n\n` between lines.

- Content: across clips, include at least one fact that is supported in one clip and contradicted in another (day/time, quantity, inclusion/exclusion). Add 1-2 neutral/distractor details.
- QAs: produce 5-7 items. For each QA:
 - uses: the minimal set of `clip_ids` needed (≥ 2 for at least half the QAs).
 - evidence_quotes: 1-2 short verbatim substrings from the referenced clips.
 - answer: a natural-language sentence that begins with "Yes, ...", "No, ...", or "Not enough information, ...", consistent with the union of the uses clips.
 - Language: English, TTS-friendly (no URLs/emojis; no sound stage directions).
 - Keep everything concise and consistent. No contradictory QAs.

Return exactly this JSON shape (no extra keys):

```

{
  "datapoint_id": "string",
  "scenario": "string",
  "clips": [
    {
      "clip_id": "A1",
      "speaker": "alloy|ash|ballad|coral|echo|fable|nova|onyx|sage|shimmer",
      "instructions": "Voice: ...\\n\\nTone: ...\\n\\nPunctuation: ...\\n\\nDelivery: ...",
      "transcript": "Plain text monologue. No timestamps. TTS-friendly."
    }
  ],
  "qa": [
    {
      "qid": "Q1",
      "question": "string",
      "uses": ["A1", "A3"],
      "evidence_quotes": [
        {"clip_id": "A1", "quote": "verbatim phrase"},
        {"clip_id": "A3", "quote": "verbatim phrase"}
      ],
      "answer": "Yes, ... / No, ... / Not enough information, ..."
    }
  ]
}

```

Checks

- Every speaker is exactly from the whitelist.
- Every quote is a verbatim substring of its referenced transcript.
- At least half of QAs require ≥ 2 clips (as listed in uses).
- answer starts with Yes, / No, / Not enough information, and matches the evidence.

Figure 7: Audio Entailment Script and QA Generation

```

864
865
866
867
868
869
870
871
872
873 I am creating a training dataset for an Audio Language Model that must handle questions relating to both the frequency and timing
874 of events in multiple audio clips.
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917

```

```

I am creating a training dataset for an Audio Language Model that must handle questions relating to both the frequency and timing
of events in multiple audio clips.

### Audio Data
You will receive data for up to five audio clips. Each clip provides:
1. **Audio Path**: A string location of the audio file.
2. **Captions**: Short descriptive statements about the sounds or events (e.g., "door slamming three times," "explosion heard
between 3 and 6 seconds," etc.).
3. **Time-Stamped Caption** and **Frequency Caption**:
   - **Time-stamped** details: Where each event starts and ends (e.g., door slamming from 0.45s-1.991s).
   - **Frequency** details: How many times a certain event occurs (e.g., "door slamming three times").

## Task Requirements
1. **Generate Frequency-Based Questions**:
   - Example: "How many times does door slamming occur in the first audio?"
   - Another Example: "Which audio has the greatest number of cow moos?"
2. **Generate Time-Stamp-Based Questions**:
   - Example: "When does the explosion start and end in audio 3?"
   - Another Example: "At which intervals do we hear door knocking in the second clip?"
3. **Combine or Compare**:
   - Feel free to combine frequency and timestamp aspects: "Which audio has the most door slams, and at what times do they occur?"
4. **Provide a Correct Answer**:
   - The answer should be a short statement or, if it's a multiple-choice format, indicate the correct choice.
   - It must reference the **actual** times, events, or frequency counts provided in the data (e.g., "It slams three times at
roughly 0.45-1.991s, 3.019-5.8s, and 6.623-8.102s.").
5. **Multiple Question-Answer Pair per Audio Set**:
   - Each set of up to five audio clips can yield **multiple** questions and answers.
   - The questions should not repeat.
   - The questions can be **MCQ** or **short-answer** type.

---
## JSON Output Format
Output your question and answer in the following **JSON** structure:
{
  "audio_paths": ["<path1>", "<path2>", ...],
  "frequencyCaption": ["<frequencyCaption1>", "<frequencyCaption2>", ...],
  "question": [<Frequency- or Time-Stamp-Based Question 1>, "<Frequency- or Time-Stamp-Based Question 2>", ...],
  "answer": [<Correct Answer 1 Grounded in the Provided Data>, "<Correct Answer 2 Grounded in the Provided Data>", ...]
}

1. audio_paths: Up to five audio file paths.
2. frequencyCaption: The descriptive or frequency-based captions for each audio.
3. questions: A list of the generated direct question.
4. answer: A list of short or MCQ correct answer that references the time-stamped or frequency data corresponding to each
question.

Example:
{
  "audio_paths": [
    "multi_event_train/syn_21.wav",
    "multi_event_train/syn_91.wav",
    "multi_event_train/syn_153.wav"
  ],
  "time_stamped_caption": ['door slamming at 0.45-1.991, 3.019-5.8, 6.623-8.102', 'door knocking at 1.155-5.305', 'door knocking at
1.155-5.305', 'cow mooing at 1.592-4.602, 6.719-9.729 and explosion at 3.329-6.882'],
  "frequencyCaption": ['door slamming three times', 'door knocking one times', 'cow mooing two times and explosion one times'],
  "question": [
    "Which audio clip has the highest frequency of door slamming, and at what timestamps do they occur?", "When does the cow moo in
the third audio?",
  ],
  "answer": [
    "Audio 1 contains door slamming three times: from about 0.45-1.991s, 3.019-5.8s, and 6.623-8.102s, which is more than any other
clip.", "Cow mooing can be heard 2 times at 1.592-4.602, 6.719-9.729"
  ]
}

Here is the audio data:

```

Figure 8: Temporal & Frequency based QA generation.

918
919
920
921
922
923
924
925
926
927

I am creating a training dataset for an Audio Language Model that can perform question answering on multiple audios. You will help me in creating the training dataset.
 You are given up to five audio clips. For each audio clip, you have the following information:
 1. Audio Path: A string representing the file's location.
 2. Caption: A descriptive caption of the content (e.g., "A dog barking in the backyard," or "Rain pouring outside"). Note that this caption does not necessarily contain a transcript of speech because these audios primarily have non-speech events.
 3. Time-Stamped Events: Labels of key acoustic events or sounds, along with their start and end times. For instance: ['(Background noise-0.000-1.144)', '(Video game sound-0.000-10.000)', '(Music-1.179-7.371)', '(Sound effect-6.375-8.045)', '(Music-9.034-10.000)'] which means the caption has "background noise from 0.000 seconds to 1.144 seconds, vide game sound from 0.000 to 10.000 seconds, music sound from 1.179 to 7.371 seconds, sound effects from 6.375 to 8.045 seconds and music sound from 9.034 to 10.000 seconds."
 Your Task:
 1. Generate questions focusing on differences among the provided audio clips. These differences can be about any audible elements such as the types of sounds, durations of events, overlaps of events, or notable acoustic aspects that might be inferred (e.g., loudness or pitch changes, even if you don't have exact numerical values).
 2. Ensure that the question:
 - Strictly compares or contrasts some aspect of at least two (or more) of the audio files.
 - Is not about any visual component or transcript-based speech recognition, but rather about the sounds present and their time-stamped events.
 - May optionally be a multiple-choice question (MCQ) or a short textual question.
 3. Generate a corresponding, correct answer:
 - This answer can be a short sentence or, in the case of MCQs, the correct option.
 - The answer should be grounded in the captions and time-stamped events.
 - Since these audios contain no explicit speech transcripts, your answer should reflect differences in non-speech audio events (e.g., "The first audio clip has a dog barking at the beginning, whereas the second has a car honking.").
 4. No Follow-Up Questions: You are to provide only one Q-A pair per set of up to five audio clips.
 Format Requirements:
 - When you output the question and answer, maintain the following json structure:
`{
 "audio_paths": ["<path1>", "<path2>", ...], // up to 5
 "captions": ["<caption1>", "<caption2>", ...],
 "question": ["<Your first generated comparison question here>", "<Your second generated comparison question here>", ...] // upto 5 questions
 "answer": ["<Correct answer for first question here>", "<Correct answer for second question here>", ...] // answers corresponding to each question
}

- audio_paths is a list of up to five paths.
- captions is a list of corresponding descriptive captions (one per audio).
- time_stamped_events is a list of lists, with each sub-list containing the labeled events for a specific audio.
- question is a list of questions that compares at least two of these clips.
- answer is a list of short sentence or the correct choice to the question.`
 Key Points:
 - Focus your question on the differences in sounds, events, or other notable audio characteristics among the clips.
 - The answer must be correct and should clearly reflect the content in the time-stamped events or captions.
 - The goal is to train an Audio Language Model that can handle multiple audios at once and answer a single comparison question about them.
 For example:
`{
 "audio_paths": ["YJ0oQj-tobYow.wav", "YSUxfKJP4bJ4.wav", "YsbY-zp5Lfew.wav", "YqExrE3FyjM.wav"],
 "captions": [
 "A man speaks with electronic music playing in the background at a martial arts gym.",
 "A sewing machine hums in the background as various tools are used intermittently in a workshop.",
 "A heavy engine starts and runs in the background noise.",
 "People are chewing and using machinery, with occasional breathing and tapping sounds."
],
 "question": ["Which audio has the heavy engine beginning after 3 seconds, and which one has vacuum noise lasting from 0 to 10 seconds?", "Between the first and fourth audios, which has continuous music from start to end, and which features chewing from around 2s to 5s?", "Comparing male speech events, which audio features multiple speech segments throughout, and which has minimal or no speech segments?"],
 "answer": [
 "The third audio starts with a heavy engine sound after about 3 seconds, while the second audio has vacuum noise for the entire 0-10s duration.",
 "The first audio clip has continuous electronic music from 0-10s, whereas the fourth audio has chewing sounds between roughly 2s and 5s.",
 "Audio 1 has multiple male speech events spread across the 0-10s window, while Audio 2 is dominated by mechanical sounds (vacuum noise) and does not have male speech segments."
]
}`
 Here are the audio details:

961
962
963
964
965
966
967
968
969
970
971

Figure 9: Audio Difference QA generation.

972

973

974

You are assisting in creating a training dataset for a Large Audio Language Model (LALM) that can answer questions about multiple music audio clips at once. You are given up to five audio clips. For each audio clip, the following metadata may be available (not all fields must be used in every question, and the LALM itself does not see these fields directly, only the raw audio):

1. name: The audio file path (e.g., `/path/to/audio.mp3`).
2. artist: The performer's name (e.g., "Cheryl").
3. song: The song title (e.g., "Rain on Me").
4. album_name: The album title.
5. tags/genres: Genres or descriptive tags (e.g., "pop", "rock," "instrumental").
6. release: The year of release (e.g., "2009").
7. danceability, energy, key, mode, valence, tempo, duration_ms: Numerical audio features that might indicate rhythmic feel, intensity, musical key/mode, emotional positivity, speed, or duration.
8. lang: The primary language of any vocals (e.g., "en" for English).

975 Possible Reasoning Types:

When generating your comparison question, pick one of the following reasoning types (or a similar audio-based angle) so that the LALM must rely on listening-based analysis. Each type is illustrated with a question example:

1. Instrumental & Vocal Presence
 - Example: "Which audio track has a more dominant presence of electric guitar throughout most of its duration?"
2. Tempo & Rhythmic Analysis
 - Example: "Which of these two tracks maintains a higher average tempo from start to finish?"
3. Tonality & Key Detection
 - Example: "Which track sounds like it is in a minor key based on its chord progression and overall tonal color?"
4. Dynamic Range & Volume Variation
 - Example: "Which track shows a wider dynamic range from its beginning to its ending section?"
5. Language & Stylistic Analysis
 - Example: "Which track features lyrics in Spanish and a rap-style vocal delivery?"
6. Genre & Stylistic Interpretation
 - Example: "Which track best fits a symphonic metal style, considering its instrumentation and overall sound?"
7. Structural Form Analysis
 - Example: "Which track follows a verse-chorus-verse structure based on repeated melodic sections?"

976 Your task is to generate a Q-A pair for each set of two audio clips. This Q-A pair should:

1. Be a comparison question between the two audio clips or a question that requires "listening" rather than just reading text (e.g., comparing energy, tempo, presence of vocals, musical complexity, or any audio-derived feature).
2. Avoid purely textual or trivial metadata lookups—questions should be framed as if the model must analyze the raw audio.
3. Demand multi-level reasoning, not a simple yes/no or single-step answer. For instance, the question might ask which track has a higher tempo and also has a more upbeat feel, or which track shifts from minor to major key, etc.
4. Only one question in every question. There should not be nested questions.
5. Provide a correct answer grounded in what the audio would contain (inferred from the metadata).
6. Format each Q-A pair using the JSON structure below.
7. Make sure to refer the audios only by their position (like 1st audio, second audio) and not by their name or any other metadata.

977 JSON Output Format

978 For each set of 2-5 audios, output a single JSON object with the following fields:

```
{
  "audio_paths": ["<audio_path_1>", "<audio_path_2>", ...],
  "metadata": [
    {
      "artist": "...",
      "song": "...",
      "tags": "...",
      "release": "...",
      "danceability": "...",
      "energy": "...",
      ...
    },
    ...
  ],
  "reasoning_type": <reasoning_type from the above list>
  "question": [<Your 1st reasoning question>, <Your 2nd reasoning question>...],
  "answer": [<Correct answer for 1st question>, <Correct answer for 2nd question>...]
}

1. audio_paths: A list of the file paths (one for each audio).
2. metadata: A list of objects, each containing relevant fields (artist, song, tags, etc.) for the corresponding audio. You can include or omit fields as necessary.
3. question: A list containing the questions string.
4. answer: A list containing the correct answers corresponding to each question in the list.
```

979 Important Guidelines

1. Comparison Focus:
 - Your question must compare or contrast at least two audios in a way that necessitates listening.
 - Examples: Differences in tempo, key, vocal presence, energy levels, or changes over time in each track.
2. No Multi-Part Questions:
 - Only one direct question. (E.g., "Which track has higher energy?" rather than "Which track has higher energy, and which one has a key change?")
 - The LALM is trained to listen, so the question should be answerable by analyzing the sound, not just reading textual metadata.
3. Audio-Centric:
 - Avoid purely text-based or metadata-based questions like "Who is the artist?" or "Which track is from 2009?"
4. Answer:
 - The answer should be concise, directly addressing the comparison.
 - Reference the audio differences logically. For instance: "Track A has a faster tempo (around 110 BPM) compared to Track B's slower beat (around 90 BPM)."
 - Visual or Speech Transcripts:
 - Stick to audio attributes like instrumentation, vocals, tempo, or mood.
 - If the track has vocals, focus on style or language, not detailed speech transcripts.

980 Example Output Snippet

981 Below is a fictional example demonstrating how to format one Q-A pair (assume we have two audios):

```
{
  "audio_paths": [
    "/path/music_track1.mp3",
    "/path/music_track2.mp3"
  ],
  "metadata": [
    {
      "artist": "Cheryl",
      "song": "Rain on Me",
      "tags": "pop,british,female vocalists",
      "release": "2009",
      "danceability": "0.635",
      "energy": "0.746"
    },
    {
      "artist": "Modestep",
      "song": "Water Thoughts",
      "tags": "instrumental hip-hop,underground hip hop",
      "release": "2013",
      "danceability": "0.591",
      "energy": "0.513"
    }
  ],
  "question": [
    "Which track has a higher overall energy level and is more likely to feature prominent vocals?"
  ],
  "answer": [
    "The first track has a higher energy score (0.746) and includes a distinct female vocal lead, whereas the second track's energy is lower (0.513) and is primarily instrumental."
  ]
}
```

982 Here is the input metadata:

983

984

985

986

987

988

989

990

991

992

993

994

995

996

997

998

999

1000

1001

1002

1003

1004

1005

1006

1007

1008

1009

1010

1011

1012

1013

1014

1015

1016

1017

1018

1019

1020

1021

1022

1023

1024

1025

Figure 10: Multi-Audio QA Generation for Music4All.