Audio-Mind: An Auditable Agentic Framework for Audio Understanding

Anonymous ACL submission

Abstract

Audio agents extend large audio-language models (LALMs) by decomposing audio questions into tool calls, intermediate evidence, and iterative reasoning steps. However, as LALMs become stronger, the key challenge shifts from enabling tool use to determining when agentic evidence acquisition genuinely benefits audio understanding. We propose AUDIO-MIND, an auditable and pluggable framework for conditional evidence acquisition in audio understanding. AUDIO-MIND dynamically combines a strong frontend with planner-guided tool use, preserving frontend judgment when initial evidence is sufficient while acquiring bounded external evidence for questions with unresolved evidence gaps. Experiments on MMAR and MSU-Bench show that AUDIO-MIND outperforms prior audio-agent baselines, reaching 80.4% accuracy on MMAR and 82.8% accuracy on MSU-Bench. A matched-backbone comparison highlights why this design matters: under strong audio frontends, agentic decomposition can become an orchestration bottleneck when the workflow does not preserve the frontend's holistic audio-grounded judgment. Beyond accuracy, AUDIO-MIND produces higher-quality, auditable reasoning traces that expose uncertainty, tool evidence, and answer rationales, offering a potential basis for more reliable audio-QA annotation and error analysis.

1 Introduction

Large audio-language models (LALMs) are moving audio understanding beyond task-specific recognition pipelines. Recent systems increasingly accept speech, environmental sounds, and music as native inputs to language models, allowing users to ask natural-language questions over diverse acoustic scenes (Peng et al., 2024; Cui et al., 2024). Recent LALM and omni-audio model families have expanded this capability across speech recognition, audio captioning, speaker and scene understanding, music analysis, and audio-grounded dialogue (Chu et al., 2024; Ding et al., 2025; Zhang et al., 2025; Huang et al., 2025; Xu et al., 2025a; Qwen Team, 2026). As these models become stronger, they provide increasingly capable frontend perception and contextual reasoning for audio understanding.

This progress has also encouraged researchers to introduce agentic mechanisms into audio understanding. Existing systems use LLM program generation to construct speech-processing toolchains, coordinate audio models or external tools for multi-step reasoning, transform audio reasoning into iterative textual evidence refinement, or learn policies for when to consult external perception (Kuan et al., 2024; Lee et al., 2025; Wijngaard et al., 2025; Rong et al., 2025; Chen et al., 2026; Tong et al., 2026; Wan et al., 2026; Xiong et al., 2025; Shao et al., 2026). Together, these studies establish audio agents as an emerging research direction for extending LALMs beyond single-pass audio question answering.

However, much of the current audio-agent literature rests on a premise that becomes less secure as LALMs improve: agentic decomposition is often expected to add useful evidence beyond direct LALM inference. This pressure is already visible in recent Qwen-Omni models which report strong performance not only on general audio understanding benchmarks, but also on ASR, speech translation, and music understanding tasks that have traditionally been served by specialized audio systems (Xu et al., 2025a; Qwen Team, 2026). **Under such a strong frontend, the value of external tools depends less on their existence than on whether their outputs are more reliable, more precise, or more targeted than the model's own perception.** This issue is even sharper because many audio properties, such as tempo or chord labels, are inherently ambiguous, convention-dependent, or only partially captured by tool outputs (Chiu et al., 2022; Koops et al., 2017). Audio-

agent systems therefore need to reason more carefully about when and how to leverage tools, rather than treating tool invocation itself as the source of improvement.

This observation motivates a shift from tool routing to conditional evidence acquisition. In this view, tools are not selected merely because they match a task name; they are invoked only when their outputs can provide bounded, task-relevant evidence that complements the frontend model. A prerequisite for such an agent is therefore a planner-facing description of what each external operation is allowed to contribute: whether it directly provides an audio observation, creates a derived artifact for later inspection, or only validates input assumptions. Making these roles explicit allows the planner to distinguish evidence from preparation, measurements from interpretations, and reliable tool outputs from signals that should remain uncertain.

To operationalize this view, we propose AUDIO-MIND, an auditable and pluggable framework for conditional evidence acquisition in audio understanding. AUDIO-MIND preserves an audio-capable LALM frontend as the main source of audio-grounded judgment, while a text-LLM planner decides whether bounded tools or targeted re-listening are needed to address a concrete evidence gap. The resulting evidence state records frontend observations, planner decisions, tool outputs, re-listening results, and final rationales, making the answer path auditable rather than treating agentic decomposition as an automatic improvement. This structure also enables higher-quality reasoning traces: by exposing uncertainty, evidence provenance, and intermediate decisions, AUDIO-MIND produces explanations that are easier to inspect, compare, and use for audio-QA error analysis.

Our contributions are summarized as follows:

- We revisit tool-augmented audio-agent reasoning under strong LALMs, showing that agentic orchestration should be treated as conditional evidence acquisition rather than an automatic improvement over end-to-end LALM inference.

- We introduce AUDIO-MIND, an auditable and pluggable agentic audio understanding framework with a reliability-aware audio-tool taxonomy and boundary design. The framework separates frontend perception, planning, tool execution, targeted re-listening, evidence fusion, and answer validation in an explicit evidence state.

- We evaluate AUDIO-MIND on MMAR and MSU-Bench, showing both accuracy gains and improved reasoning traces. AUDIO-MIND outperforms prior audio-agent baselines, while matched-backbone comparisons show that existing agentic workflows can become orchestration bottlenecks when they fail to preserve strong frontend perception. Beyond final-answer accuracy, AUDIO-MIND produces more auditable and higher-quality reasoning traces for audio-QA inspection and error analysis.

2 Related Work

2.1 Large Audio-Language Models and Audio Understanding Benchmarks

Recent work on speech and audio language models has moved from task-specific pipelines toward general-purpose models that accept audio as a native input modality (Peng et al., 2024; Cui et al., 2024). Models such as SALMONN (Tang et al., 2023), Qwen2-Audio (Chu et al., 2024), Kimi-Audio (Ding et al., 2025), MiMo-Audio (Zhang et al., 2025), Step-Audio (Huang et al., 2025), and Audio Flamingo (Kong et al., 2024; Goel et al., 2025) extend language-model interfaces to speech, sounds, music, and audio-grounded dialogue. More recent omni or reasoning-oriented systems, including Qwen3-Omni (Xu et al., 2025a), Qwen3.5-Omni (Qwen Team, 2026), and Step-Audio-R1 (Tian et al., 2025), further strengthen audio perception, interaction, and modality-grounded reasoning. These models define increasingly strong frontend baselines for evaluating whether audio-agent mechanisms add useful evidence beyond LALM inference.

Evaluation has likewise moved beyond isolated recognition tasks. Recent benchmarks, including AIR-Bench (Yang et al., 2024), MMAU (Sakshi et al., 2024), MMAR (Ma et al., 2025), MMAU-Pro (Kumar et al., 2025), and MSU-Bench (Wang et al., 2025), test increasingly heterogeneous forms of audio understanding, ranging from instruction following and audio reasoning to long-form, multi-audio, spatial, open-ended, and multi-talker scenarios. These benchmarks make strong LALM baselines essential, while also motivating systems that

can expose intermediate evidence across diverse audio tasks.

2.2 Tool-Augmented and Agentic Audio Understanding

Building on general tool-using LLM agents that interleave reasoning with external actions (Yao et al., 2022), recent audio-agent systems adapt this paradigm to audio understanding. These systems explore different ways to integrate these tools with LALMs. Speech-Copilot (Kuan et al., 2024) decomposes speech-processing instructions and generates programs over modular tools, while AudioToolAgent (Wijngaard et al., 2025) coordinates audio-language models and tool adapters through a central LLM agent. Audio-Maestro (Lee et al., 2025) and AudioGenie-Reasoner (Rong et al., 2025) emphasize tool-augmented or multi-agent evidence construction for audio-language reasoning. Other work studies tool-use policy itself: AudioRouter (Chen et al., 2026) and AuTAgent (Tong et al., 2026) learn when and how to invoke tools through reinforcement learning, Speech-Hands (Wan et al., 2026) frames external audio perception as a self-reflection decision, and Multi-Source Evidence Fusion (Olev and Alumäe, 2026) combines independent LALM observations with acoustic tools organized by reliability tiers. These works establish the promise of agentic audio understanding. However, they primarily study how to introduce or optimize external evidence use, while our work asks when such evidence should be acquired under strong audio frontends and treat tool use as conditional evidence acquisition rather than an automatic improvement over direct LALM inference.

Adjacent systems also show that intermediate audio operations can support understanding even when they are not framed as general-purpose tool agents. Thinking with Sound (Xiong et al., 2025) combines linguistic reasoning with on-the-fly audio-domain analysis and manipulation, while Listening with Time (Shao et al., 2026) uses global-to-local temporal reasoning with local audio evidence for long-form understanding. Relative to these systems, our framework makes the roles of such intermediate operations more explicit by distinguishing direct evidence extraction from audio transformation, and by exposing both through capability-bounded interfaces.

3 Tools as Bounded Evidence Interfaces

Before presenting the full agent architecture, we first characterize the role of tool use in audio-agent reasoning. Audio tools play a different role in audio agents than tools usually play in language-only agents. Rather than only expanding the model's external action space, they often operate on the input signal itself: exposing a partial observation of the audio or reshaping the audio so that a later model or tool can inspect it more effectively. Therefore, we treat audio tools as evidence interfaces rather than unconditional substitutes for the frontend LALM.

3.1 Perception and Transformation Roles

Under this view, we distinguish two broad roles. *Perception tools* extract observations from audio, such as transcripts, speaker turns, temporal boundaries, pitch, key, chords, or spectral measurements. These tools can support an answer when the question depends on a local, symbolic, or measurable cue. *Transformation tools* modify the audio source itself, for example by trimming, denoising, separating, filtering, resampling, or converting channels. Their output is not usually evidence by itself; rather, it creates a better input for later perception by a tool or by the frontend model during targeted re-listening.

This distinction matters because not every tool output should be interpreted in the same way. A transcript, segment boundary, acoustic measurement, or derived audio clip may provide strong support for some questions, but may become misleading if the planner treats it as evidence for a broader judgment than it can justify.

3.2 Tool Boundaries

Tool boundaries specify what kind of claim a tool output is allowed to support. This is necessary because the same low-level observation can be useful for one question and misleading for another. For example, a transcript can verify lexical content but not speaker identity, while a tempo, chord, or spectral estimate can provide measurable acoustic evidence but not a complete music-theoretic or perceptual judgment. Without such boundaries, an agent may convert a narrow measurement into an unsupported semantic conclusion simply because the output appears precise.

These constraints should not be read as an argument against specialized tools. Rather, boundaries make tool evidence usable by specifying the condi-

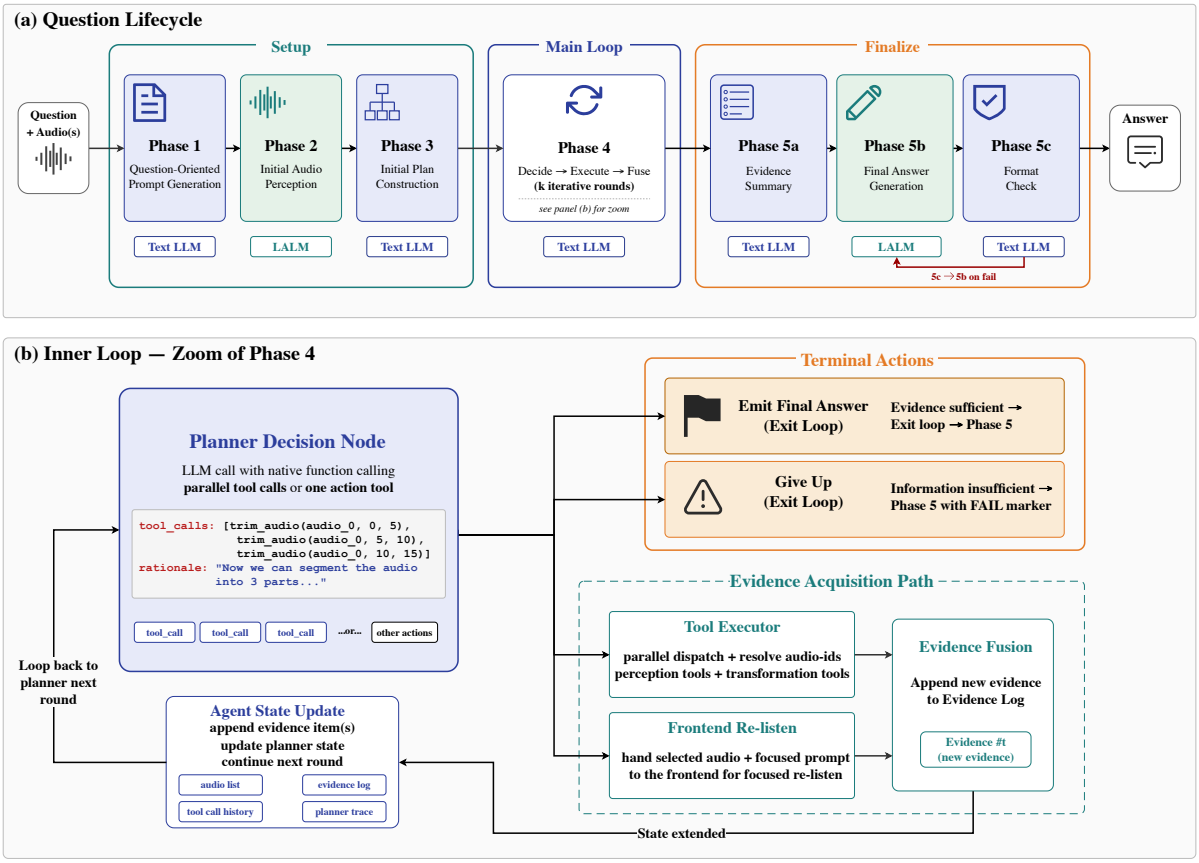

Figure 1: Architecture of AUDIO-MIND. The system first uses a planner-generated perception prompt to obtain question-oriented frontend evidence, then iteratively gathers additional evidence through tool calls or targeted frontend follow-up. Once the planner determines that the evidence is sufficient, the frontend model generates the final answer from the original audio and accumulated context, followed by format validation.

tions under which it should be trusted. Even when strong LALMs achieve higher aggregate scores on some audio benchmarks, a frontend LALM is optimized for broad audio-language behavior, while many tools are optimized for narrower interfaces and target scenes with explicit outputs. ASR and diarization systems, for example, may remain more dependable in matched settings such as meetings, interviews, or multi-speaker recordings, where their training objectives and output schemas directly target the needed evidence (Shi et al., 2026; Xu et al., 2025b; Han et al., 2025; Park et al., 2025). These outputs can be easier to inspect, compare, reuse, and pass to later stages than a free-form model response. We therefore treat tool use as conditional evidence acquisition: a tool is valuable when its bounded output provides verifiable support that the frontend either lacks, states uncertainly, or cannot express in the required structure.

4 Framework Architecture

AUDIO-MIND operationalizes conditional evidence acquisition with three design goals. First, it should preserve the frontend LALM when direct audio perception is already sufficient. Second, it should expose the unresolved evidence gap before invoking tools. Third, it should make each external operation bounded and inspectable, so that tool outputs, transformed audio artifacts, and re-listening results can be traced back to the decision that requested them.

Figure 1 illustrates the architecture of AUDIO-MIND. The system is organized as a stateful graph in which each module reads a shared evidence state, performs one bounded operation, and writes structured updates for later modules.

4.1 System Overview

Given audio inputs $\mathcal{A}$ and a natural-language question q, the system aims to produce an answer y that is grounded in the audio and satisfies the expected

Category	Description	Representative tools
	Transformation tools	
Audio derivation (8)	Creates derived audio by trimming, denoising, resampling, conversion, or separation to prepare better inputs for later tools or re-listening.	Time-range trimming; denoising; channel conversion
Temporal segmentation (5)	Locates candidate speech, silence, event, or music regions when relevant time spans or boundaries need verification.	Audio segmentation; silence detection; VAD
	Perception tools	
Metadata validation (2)	Checks file-level duration, sampling, channel, and format facts to validate input assumptions rather than audio content.	Audio stream statistics; loading metadata
Speech and speaker processing (9)	Verifies speech content, speaker counts, turns, or identity using ASR, diarization, VAD, and speaker verification.	ASR; diarization; speaker verification
Acoustic/music feature analysis (12)	Provides supporting measurements such as pitch, spectrum, tempo, key, or chords.	Pitch analysis; tempo estimation; chord recognition
Signal visualization inspection (2)	Inspects generated waveforms, spectrograms, or processing artifacts for bounded signal-level cues.	Audio-plot inspection; quality verification

Table 1: Planner-facing tool categories used by AUDIO-MIND. Numbers in parentheses indicate the number of planner-visible core tools in each category.

output format. The agent is organized around an explicit evidence state that records audio artifacts, frontend observations, planner decisions, tool outputs, evidence summaries, and format feedback. Each graph node reads this state, performs one bounded operation, and writes structured updates for later stages.

In our default instantiation, AUDIO-MIND separates audio perception from textual orchestration. An audio-capable LALM serves as the frontend for initial perception, targeted re-listening, and final answer generation, while a text LLM serves as the planner for prompt construction, action selection, evidence summarization, and format checking. The shared evidence state keeps these roles pluggable: frontends, planners, tool executors, and checkers can be replaced as long as they preserve the same evidence contracts.

4.2 Question-Oriented Perception and Planning

The first stage is designed to avoid both generic captioning and premature tool use. Directly prompting the frontend with the original question can produce an overcommitted answer or omit the evidence needed for later verification. AUDIO-MIND instead asks the planner to generate a self-contained perception prompt that restates the question, names task-specific listening targets, and asks the frontend to report uncertainty. The frontend analyzes $\mathcal{A}$ under this prompt and returns structured evidence: a general description, relevant focus points, a preliminary answer when available, and uncertainty or verification needs.

The planner treats the frontend caption as evidence rather than ground truth. It converts the caption into a lightweight plan that records the clarified intent, expected answer format, focus points, and candidate evidence operations. The plan does not execute tools; it only marks which claims appear answerable by direct perception and which may require verification.

Both the caption and the plan are stored in the shared state. Each input audio file receives a stable identifier such as `audio_0`, and any derived artifact created later is appended to the same audio list with its own identifier and provenance. This lets later decisions refer to original or transformed audio consistently while keeping the evidence path inspectable.

4.3 Evidence Acquisition Loop

The evidence acquisition loop is the point where the agent decides whether additional evidence is worth obtaining. At each iteration, the planner observes the current evidence log, tool-call history, planner trace, available audio artifacts, and available tool inventory. It then emits exactly one action type: call one or more independent tools, request frontend follow-up on selected audio, proceed to

final answer generation, or stop with an explicit failure. Each decision must include a short rationale, which is stored in the planner trace and shown back to the planner in later rounds. This makes tool use accountable to a stated evidence gap rather than to tool availability alone.

External tools are exposed through the category-bounded inventory in Table 1. The inventory turns the tool-boundary discussion in Section 3 into planner-facing guidance, so action selection depends on what evidence a tool can support rather than on tool names alone. Tool execution returns either an evidence item, a derived audio artifact, or both. This keeps analysis results and transformed inputs in the same shared state, allowing later decisions to use tool outputs without exposing low-level execution details.

Frontend follow-up, or re-listening, is used when the unresolved question is better answered by direct perception on a better-selected input. This is especially useful after tools create a materially improved artifact, such as a trimmed segment containing the relevant event, or a separated speaker track that makes the target source easier to hear. The planner sends selected audio identifiers and a focused prompt to the frontend, and the resulting caption is appended as follow-up evidence. The system does not use re-listening as a generic fallback; it is reserved for cases where selection or transformation changes what the frontend can perceive.

4.4 Answer Generation and Validation

The final stage keeps answer synthesis separate from evidence acquisition. Once the planner judges that the evidence is sufficient, it emits an answer action rather than writing the final response itself. The system then prepares a neutral evidence summary that reports observations, preserves apparent contradictions, and avoids deciding which source is correct. The audio-capable frontend receives this summary together with the original audio and task context, so the final answer is still grounded in direct listening while being informed by accumulated evidence.

A separate format checker then validates only whether the proposed answer follows the expected structure. It does not judge content correctness; it simply returns structural feedback when the answer format is invalid. This prevents formatting errors from being conflated with evidence failures.

5 Results

5.1 Experimental Setup

We evaluate AUDIO-MIND on two audio understanding benchmarks: MMAR (Ma et al., 2025) and MSU-Bench (Wang et al., 2025). MMAR covers a broad range of audio question-answering categories, while MSU-Bench focuses on information-dense multi-speaker understanding.

For each benchmark, we use the strongest accessible LALM as the audio-capable frontend. On MMAR, AUDIO-MIND uses Qwen3.5-Omni as the frontend model and Qwen3.5 as the text planner. On MSU-Bench, AUDIO-MIND uses Gemini 2.5 Pro as the frontend model and Gemini 3.0 Flash as the text planner. We also evaluate the corresponding LALM baseline, where the audio-capable model answers the benchmark question without agentic tool use.

We compare against AudioGenie-Reasoner as an agentic baseline (Rong et al., 2025). Our main purpose is to test how its agentic design behaves when paired with a strong LALM frontend, so we instantiate AudioGenie-Reasoner with the matched backbone for each benchmark: Qwen3.5-Omni and Qwen3.5 on MMAR, and the Gemini configuration on MSU-Bench. All systems are evaluated with the official multiple-choice answer format of each benchmark, and predictions are scored by matching the selected option against the ground-truth answer.

5.2 Overall Results

Tables 2 and 3 report the main benchmark results. On MMAR, AUDIO-MIND reaches 80.40 average accuracy, improving over our Qwen3.5-Omni evaluation at 78.90 while also outperforming the prior audio-agent baselines in the table. On MSU-Bench, AUDIO-MIND reaches 0.828 accuracy, slightly above the direct Gemini 2.5 Pro counterpart at 0.819 and the matched-setting AudioGenie-Reasoner result at 0.789. These gains over strong LALM counterparts are modest, but they are consistent with the intended design: the agent preserves strong frontend perception as the final audio-grounded decision point while selectively acquiring external evidence when the planner identifies a concrete uncertainty or verification need. The MMAR sub-category results further suggest where this design is most useful. AUDIO-MIND shows larger gains on Music Theory and Temporal Analysis, where structured music measurements and localized or transformed audio seg-

Method	Sound	Music	Speech	So-Mu	So-Sp	Mu-Sp	So-Mu-Sp	Avg.
End-to-end models								
GPT-4o Audio (Hurst et al., 2024)	53.94	50.97	70.41	63.64	72.48	62.20	75.00	63.50
Gemini 2.0 Flash (Google, 2025)	61.21	50.97	72.11	81.82	72.48	65.85	70.83	65.60
Gemini 2.5 Pro (Comanici et al., 2025)	67.30	56.80	82.00	**100.00**	84.90	80.50	66.70	74.70
Gemini 3 Pro (Google DeepMind, 2026)	72.73	61.65	84.35	72.73	81.19	78.05	83.33	76.40
Qwen3.5-Omni (Qwen Team, 2026)	–	–	–	–	–	–	–	80.00
Qwen3.5-Omni† (Qwen Team, 2026)	75.76	61.17	85.37	90.91	**87.61**	79.27	**87.50**	78.90
Audio agents								
AudioRouter (Chen et al., 2026)	61.40	**66.10**	77.20	57.20	65.10	68.30	55.50	64.40
AuTAgent (Tong et al., 2026)	51.52	45.15	73.81	–	–	–	–	63.00
AudioToolAgent (Wijngaard et al., 2025)	73.33	63.59	82.99	63.64	83.03	**81.71**	79.17	77.00
AudioGenie-Reasoner (Rong et al., 2025)	49.68	43.26	69.23	45.45	64.53	65.33	59.09	58.85
AudioGenie-Reasoner† (Matched Backbone) (Rong et al., 2025)	70.91	54.85	79.59	54.55	72.48	76.83	58.33	70.50
AUDIO-MIND†	**78.18**	63.11	**87.41**	**100.00**	**87.61**	79.27	**87.50**	**80.40**

Table 2: MMAR accuracy by modality. So, Mu, and Sp denote sound, music, and speech. Rows marked with † are evaluated by us on the same 1,000 MMAR examples; other values are taken from the corresponding benchmark or system reports.

Model	Avg.
End-to-end models	
Kimi-Audio (Ding et al., 2025)	0.460
StepAudio2 (Wu et al., 2025)	0.480
MiMoAudio (Zhang et al., 2025)	0.600
Gemini 2.5 Flash† (Comanici et al., 2025)	0.682
Qwen3.5-Omni† (Qwen Team, 2026)	0.763
Gemini 2.5 Pro (thinking)† (Comanici et al., 2025)	0.819
Agentic systems	
AudioGenie-Reasoner† (Matched Backbone) (Rong et al., 2025)	0.789
AUDIO-MIND†	**0.828**

Table 3: Average accuracy on MSU-Bench. Rows marked with † are evaluated by us; unmarked values are reported by MSU-Bench (Wang et al., 2025). AudioGenie-Reasoner and AUDIO-MIND use the Gemini configuration for MSU-Bench.

ments can provide evidence that is difficult for a single-pass frontend response to retain precisely. Appendix A provides the full MMAR sub-category breakdown for Qwen3.5-Omni, matched-backbone AudioGenie-Reasoner, and AUDIO-MIND.

The matched-backbone AudioGenie-Reasoner result illustrates why stronger base models do not automatically make an agentic pipeline stronger. On MMAR, AudioGenie-Reasoner instantiated with Qwen3.5-Omni and Qwen3.5 rises substantially over its originally reported result (70.50 vs. 58.85 average accuracy), showing that the stronger backbone helps. However, it still falls noticeably below Qwen3.5-Omni in our evaluation on MMAR (70.50 vs. 78.90). On MSU-Bench, using the same Gemini setting as AUDIO-MIND, AudioGenie-Reasoner reaches 0.789, below both Gemini 2.5 Pro direct inference at 0.819 and AUDIO-MIND at 0.828. This suggests that some agentic designs can underuse a strong omni frontend: if the audio model is mainly treated as an intermediate evidence generator, and the final decision is routed through compressed textual traces, the pipeline may lose part of the holistic perception and reasoning ability already available in the frontend model. Thus, agentic audio reasoning should not be viewed as a guaranteed improvement over LALM inference; its benefit depends on whether the agent preserves reliable frontend evidence while adding useful external evidence.

5.3 Behavioral Analysis of Tool Use

To examine whether AUDIO-MIND behaves the way conditional evidence acquisition predicts, we stratify the 999 successfully completed MMAR runs by the number of tool calls the planner makes per question and compare against the Qwen3.5-Omni baseline on the same items. Across all examples the planner uses on average 1.68 tool calls per question, and 31.2% of questions exit at zero tool calls. The framework is therefore empirically conditional rather than always-on.

Table 4 reveals two coupled trends. First, Qwen3.5-Omni accuracy decreases monotonically as tool-call count grows, from 90.4% at zero calls to 52.0% in the 6–10 bucket. Since the planner has no access to ground-truth labels or baseline outcomes, and only observes the question, frontend-generated evidence, accumulated state, and available tool interfaces, this alignment suggests that AUDIO-MIND captures a practical, question-level estimate of the frontend LALM's capability boundary through its evidence-gap formulation: tool-use depth increases when the initial frontend evidence appears insufficient and remains low when it ap-

Tool calls	N	AUDIO-MIND	Qwen3.5-Omni	Δ
0	311	89.7	90.4	-0.6
1	255	85.5	86.3	-0.8
2	164	73.8	72.6	+1.2
3	142	72.5	66.2	+6.3
4–5	96	69.8	60.4	+9.4
6–10	25	56.0	52.0	+4.0
>10	6	16.7	50.0	-33.3

Table 4: MMAR accuracy comparison between AUDIO-MIND and Qwen3.5-Omni, grouped by the number of tool calls made by AUDIO-MIND. Accuracy and Δ are reported in percentage points.

pears adequate. Thus, conditional evidence acquisition is reflected in the agent's behavior, not only in its design. Second, the AUDIO-MIND–Qwen3.5-Omni gap becomes positive in higher tool-use buckets. With zero or one tool call, AUDIO-MIND remains close to direct inference ($\Delta \in [-0.8, -0.6]$), suggesting that it largely preserves frontend performance when little additional evidence is needed. Gains emerge from two calls onward, reaching +9.4 points in the 4–5 call bucket, indicating that tool-use benefits are concentrated in cases where the planner commits to deeper evidence acquisition rather than uniformly improving all questions.

Together, these trends help explain the aggregate $+1.5$-point improvement: AUDIO-MIND remains close to direct frontend inference when the planner requests little or no tool evidence, while its gains concentrate on questions that trigger deeper evidence acquisition, which is consistent with the intended behavior of conditional evidence acquisition. Appendix B reports planner-round and tool-frequency statistics, including the long-tail > 10-call bucket.

5.4 Reasoning Trace Quality Beyond Accuracy

AUDIO-MIND is designed to make the reasoning process more interpretable by construction. A LALM answer usually compresses perception, reasoning, and justification into a single response, making it difficult to tell which audio cues were actually heard, which uncertainties were ignored, and whether the explanation is faithful to the decision. In contrast, AUDIO-MIND separates the trace into explicit stages. This structure does not guarantee correctness, but it makes the answer path easier to audit because unsupported assumptions, conflicting evidence, and tool-dependent claims remain visible.

For reasoning-trace quality, we apply MMAR-Rubrics, the official instance-level protocol from the Interspeech 2026 Audio Reasoning Challenge (Ma et al., 2026). It scores each correct-answer trace by averaging five instance-specific binary criteria and assigns zero to incorrect answers. Under this protocol, AUDIO-MIND obtains an MMAR-Rubrics score of 66.5%, compared with 59.6% for Qwen3.5-Omni.[1] This result suggests that AUDIO-MIND can provide higher-quality audio-QA reasoning chains, which may be valuable for downstream data annotation, error analysis, and human inspection beyond aggregate benchmark accuracy.

Appendix C provides a qualitative example of how such traces support inspection. The case illustrates three recurring behaviors: an LALM may hallucinate a plausible acoustic rationale, the question-oriented frontend prompt can expose uncertainty, and bounded tools can provide the missing evidence. This example complements the MMAR-Rubrics results by showing how auditable traces make the correction process visible rather than only reporting a final answer.

6 Conclusion

We presented AUDIO-MIND, an auditable and pluggable framework for audio understanding under strong LALM frontends. Motivated by the observation that agentic decomposition is not automatically beneficial when LALMs become stronger, AUDIO-MIND treats tool use as conditional evidence acquisition. It preserves frontend audio-grounded judgment while selectively using bounded tools and targeted re-listening when additional evidence is needed.

Experiments on MMAR and MSU-Bench show that AUDIO-MIND outperforms prior audio-agent baselines and produces more auditable reasoning traces. Behavioral analysis further shows that AUDIO-MIND turns conditional evidence acquisition into observable tool allocation, with gains concentrated where additional evidence is most needed. These results suggest that the value of audio agents lies not in more tool calls, but in acquiring the right evidence while keeping uncertainty, tool outputs, re-listening decisions, and final rationales inspectable. This makes AUDIO-MIND a useful basis for reliable audio-QA reasoning, annotation, and error analysis.

[1] Scores are not directly comparable to the challenge report because we use Qwen3.5 as the judge instead of GPT-4o.

Limitations

Our evaluation is limited to MMAR and MSU-Bench, which cover broad audio-QA and multi-speaker understanding but do not fully represent real-world uses such as long-form meeting analysis, audio editing, forensic listening, or interactive inspection. AUDIO-MIND also depends on the quality of external tools: although tool outputs are treated as bounded evidence rather than ground truth, they may still be noisy, domain-limited, or convention-dependent, especially for music and acoustic analysis. Finally, the framework introduces additional computation through planning, tool execution, and targeted re-listening, so direct LALM inference may remain preferable for simple or latency-sensitive questions.

Ethical Considerations

This work uses existing benchmarks, pretrained models, and external audio tools for research evaluation only. We do not collect new human-subject data or recruit human annotators, and we do not redistribute third-party model weights, benchmark data, or tool artifacts. Since audio-agent systems may be used in sensitive speech or speaker-related settings, outputs from AUDIO-MIND should be treated as decision support rather than ground truth, especially for personal or legally consequential audio.

References

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, and 1 others. 2025. Qwen3-vl technical report. *arXiv preprint arXiv:2511.21631*.

Max Bain, Jaesung Huh, Tengda Han, and Andrew Zisserman. 2023. Whisperx: Time-accurate speech transcription of long-form audio. In *Proceedings of Interspeech 2023*.

Hervé Bredin. 2023. pyannote.audio 2.1 speaker diarization pipeline: Principle, benchmark, and recipe. In *Proceedings of Interspeech 2023*.

Liyang Chen, Hongkai Chen, Yujun Cai, Sifan Li, Qingwen Ye, and Yiwei Wang. 2026. Audiorouter: Data efficient audio understanding via rl based dual reasoning. *arXiv preprint arXiv:2602.10439*.

Ching-Yu Chiu, Meinard Müller, Matthew EP Davies, Alvin Wen-Yu Su, and Yi-Hsuan Yang. 2022. An analysis method for metric-level switching in beat tracking. *IEEE Signal Processing Letters*, 29:2153–2157.

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

Gheorghe Comanici, Eric Bieber, Mike Schaekermann, Ice Pasupat, Noveen Sachdeva, Inderjit Dhillon, Marcel Blistein, Ori Ram, Dan Zhang, Evan Rosen, and 1 others. 2025. Gemini 2.5: Pushing the frontier with advanced reasoning, multimodality, long context, and next generation agentic capabilities. *arXiv preprint arXiv:2507.06261*.

Wenqian Cui, Dianzhi Yu, Xiaoqi Jiao, Ziqiao Meng, Guangyan Zhang, Qichao Wang, Yiwen Guo, and Irwin King. 2024. Recent advances in speech language models: A survey. *arXiv preprint arXiv:2410.03751*.

Ding Ding, Zeqian Ju, Yichong Leng, Songxiang Liu, Tong Liu, Zeyu Shang, Kai Shen, Wei Song, Xu Tan, Heyi Tang, and 1 others. 2025. Kimi-audio technical report. *arXiv preprint arXiv:2504.18425*.

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

Google. 2025. Gemini 2.0 flash model card.

Google DeepMind. 2026. Gemini 3.

Jiangyu Han, Petr Pálka, Marc Delcroix, Federico Landini, Johan Rohdin, Jan Cernocký, and Lukáš Burget. 2025. Efficient and generalizable speaker diarization via structured pruning of self-supervised models. *arXiv preprint arXiv:2506.18623*.

Ailin Huang, Boyong Wu, Bruce Wang, Chao Yan, Chen Hu, Chengli Feng, Fei Tian, Feiyu Shen, Jingbei Li, Mingrui Chen, and 1 others. 2025. Step-audio: Unified understanding and generation in intelligent speech interaction. *arXiv preprint arXiv:2502.11946*.

Aaron Hurst, Adam Lerer, Adam P Goucher, Adam Perelman, Aditya Ramesh, Aidan Clark, AJ Ostrow, Akila Welihinda, Alan Hayes, Alec Radford, and 1 others. 2024. Gpt-4o system card. *arXiv preprint arXiv:2410.21276*.

Junyan Jiang, Ke Chen, Wei Li, and Gus Xia. 2019. Large-vocabulary chord transcription via chord structure decomposition. In *Proceedings of the 20th International Society for Music Information Retrieval Conference*, pages 644–651, Delft, The Netherlands.

Zhifeng Kong, Arushi Goel, Rohan Badlani, Wei Ping, Rafael Valle, and Bryan Catanzaro. 2024. Audio flamingo: A novel audio language model with few-shot learning and dialogue abilities. *arXiv preprint arXiv:2402.01831*.

Hendrik Vincent Koops, W Bas de Haas, Jeroen Bransen, and Anja Volk. 2017. Chord label personalization through deep learning of integrated harmonic interval-based representations. *arXiv preprint arXiv:1706.09552*.

Chun-Yi Kuan, Chih-Kai Yang, Wei-Ping Huang, Ke-Han Lu, and Hung-yi Lee. 2024. Speech-copilot: Leveraging large language models for speech processing via task decomposition, modularization, and program generation. *arXiv preprint arXiv:2407.09886*.

Sonal Kumar, Šimon Sedláček, Vaibhavi Lokegaonkar, Fernando López, Wenyi Yu, Nishit Anand, Hyeonggon Ryu, Lichang Chen, Maxim Plička, Miroslav Hlaváček, and 1 others. 2025. MMAU-Pro: A challenging and comprehensive benchmark for holistic evaluation of audio general intelligence. *arXiv preprint arXiv:2508.13992*.

Kuan-Yi Lee, Tsung-En Lin, and Hung-Yi Lee. 2025. Audio-maestro: Enhancing large audio-language models with tool-augmented reasoning. *arXiv preprint arXiv:2510.11454*.

Ziyang Ma, Yinghao Ma, Yanqiao Zhu, Chen Yang, Yi-Wen Chao, Ruiyang Xu, Wenxi Chen, Yuanzhe Chen, Zhuo Chen, Jian Cong, and 1 others. 2025. MMAR: A challenging benchmark for deep reasoning in speech, audio, music, and their mix. *arXiv preprint arXiv:2505.13032*.

Ziyang Ma, Ruiyang Xu, Yinghao Ma, Chao-Han Huck Yang, Bohan Li, Jaeyeon Kim, Jin Xu, Jinyu Li, Carlos Busso, Kai Yu, Eng Siong Chng, and Xie Chen. 2026. The interspeech 2026 audio reasoning challenge: Evaluating reasoning process quality for audio reasoning models and agents. *arXiv preprint arXiv:2602.14224*.

Brian McFee, Colin Raffel, Dawen Liang, Daniel PW Ellis, Matt McVicar, Eric Battenberg, Oriol Nieto, and 1 others. 2015. librosa: Audio and music signal analysis in python. *SciPy*, 2015(18-24):7.

Aivo Olev and Tanel Alumäe. 2026. Multi-source evidence fusion for audio question answering. *arXiv preprint arXiv:2603.17822*.

Taejin Park, Ivan Medennikov, Kunal Dhawan, Weiqing Wang, He Huang, Nithin Rao Koluguri, Krishna C. Puvvada, Jagadeesh Balam, and Boris Ginsburg. 2025. Sortformer: A novel approach for permutation-resolved speaker supervision in speech-to-text systems. In *Proceedings of the 42nd International Conference on Machine Learning*.

Jing Peng, Yucheng Wang, Bohan Li, Yiwei Guo, Hankun Wang, Yangui Fang, Yu Xi, Haoyu Li, Xu Li, Ke Zhang, Shuai Wang, and Kai Yu. 2024. A survey on speech large language models for understanding. *arXiv preprint arXiv:2410.18908*.

Alexis Plaquet and Hervé Bredin. 2023. Powerset multi-class cross entropy loss for neural speaker diarization. In *Proceedings of Interspeech 2023*.

Qwen Team. 2026. Qwen3.5-omni technical report. *arXiv preprint arXiv:2604.15804*.

Yan Rong, Chenxing Li, Dong Yu, and Li Liu. 2025. Audiogenie-reasoner: A training-free multi-agent framework for coarse-to-fine audio deep reasoning. *arXiv preprint arXiv:2509.16971*.

S. Sakshi, Utkarsh Tyagi, Sonal Kumar, Ashish Seth, Ramaneswaran Selvakumar, Oriol Nieto, Ramani Duraiswami, Sreyan Ghosh, and Dinesh Manocha. 2024. MMAU: A massive multi-task audio understanding and reasoning benchmark. *arXiv preprint arXiv:2410.19168*.

Hendrik Schreiber and Meinard Müller. 2018. A single-step approach to musical tempo estimation using a convolutional neural network. In *Proceedings of the 19th International Society for Music Information Retrieval Conference*, pages 98–105, Paris, France.

Mingchen Shao, Hang Su, Wenjie Tian, Bingshen Mu, Zhennan Lin, Lichun Fan, Zhenbo Luo, Jian Luan, and Lei Xie. 2026. Listening with time: Precise temporal awareness for long-form audio understanding. *arXiv preprint arXiv:2604.22245*.

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. 2026. Qwen3-asr technical report. *arXiv preprint arXiv:2601.21337*.

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

Fei Tian, Xiangyu Tony Zhang, Yuxin Zhang, Haoyang Zhang, Yuxin Li, Daijiao Liu, Yayue Deng, Donghang Wu, Jun Chen, Liang Zhao, Chengyuan Yao, Hexin Liu, Eng Siong Chng, Xuerui Yang, Xiangyu Zhang, Daxin Jiang, and Gang Yu. 2025. Step-audio-r1 technical report. *arXiv preprint arXiv:2511.15848*.

Suramya Tomar. 2006. Converting video formats with ffmpeg. *Linux journal*, 2006(146):10.

Siqian Tong, Xuan Li, Yiwei Wang, Baolong Bi, Yujun Cai, Shenghua Liu, Yuchen He, and Chengpeng Hao. 2026. Autagent: A reinforcement learning framework for tool-augmented audio reasoning. *arXiv preprint arXiv:2602.13685*.

Zhen Wan, Chao-Han Huck Yang, Jinchuan Tian, Hanrong Ye, Ankita Pasad, Szu-wei Fu, Arushi Goel, Ryo Hachiuma, Shizhe Diao, Kunal Dhawan, Sreyan Ghosh, Yusuke Hirota, Zhehuai Chen, Rafael Valle, Ehsan Hosseini Asl, Chenhui Chu, Shinji Watanabe, Yu-Chiang Frank Wang, and Boris Ginsburg. 2026. Speech-hands: A self-reflection voice agentic approach to speech recognition and audio reasoning with omni perception. *arXiv preprint arXiv:2601.09413*.

Hongji Wang, Chengdong Liang, Shuai Wang, Zhengyang Chen, Binbin Zhang, Xu Xiang, Yanlei Deng, and Yanmin Qian. 2023. Wespeaker: A research and production oriented speaker embedding learning toolkit. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing*.

Shuai Wang, Zhaokai Sun, Zhennan Lin, Chengyou Wang, Zhou Pan, and Lei Xie. 2025. MSU-Bench: Towards understanding the conversational multi-talker scenarios. *arXiv preprint arXiv:2508.08155*.

Gijs Wijngaard, Elia Formisano, Michel Dumontier, and Jenia Jitsev. 2025. Audiotoolagent: An agentic framework for audio-language models. *arXiv preprint arXiv:2510.02995*.

Boyong Wu, Chao Yan, Chen Hu, Cheng Yi, Chengli Feng, Fei Tian, Feiyu Shen, Gang Yu, Haoyang Zhang, Jingbei Li, and 1 others. 2025. Step-audio 2 technical report. *arXiv preprint arXiv:2507.16632*.

Zhen Xiong, Yujun Cai, Zhecheng Li, Junsong Yuan, and Yiwei Wang. 2025. Thinking with sound: Audio chain-of-thought enables multimodal reasoning in large audio-language models. *arXiv preprint arXiv:2509.21749*.

Jin Xu, Zhifang Guo, Hangrui Hu, Yunfei Chu, Xiong Wang, Jinzheng He, Yuxuan Wang, Xian Shi, Ting He, Xinfa Zhu, and 1 others. 2025a. Qwen3-omni technical report. *arXiv preprint arXiv:2509.17765*.

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

Kaituo Xu, Yan Jia, Kai Huang, Junjie Chen, Wenpeng Li, Kun Liu, Feng-Long Xie, Xu Tang, and Yao Hu. 2026. Fireredasr2s: A state-of-the-art industrial-grade all-in-one automatic speech recognition system. *arXiv preprint arXiv:2603.10420*.

Qian Yang, Jin Xu, Wenrui Liu, Yunfei Chu, Ziyue Jiang, Xiaohuan Zhou, Yichong Leng, Yuanjun Lv, Zhou Zhao, Chang Zhou, and Jingren Zhou. 2024. AIR-Bench: Benchmarking large audio-language models via generative comprehension. *arXiv preprint arXiv:2402.07729*.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2022. ReAct: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*.

Dong Zhang, Gang Wang, Jinlong Xue, Kai Fang, Liang Zhao, Rui Ma, Shuhuai Ren, Shuo Liu, Tao Guo, Weiji Zhuang, and 1 others. 2025. Mimo-audio: Audio language models are few-shot learners. *arXiv preprint arXiv:2512.23808*.

A MMAR Sub-Category Results

Table 5 reports the MMAR sub-category comparison among Qwen3.5-Omni, matched-backbone AudioGenie-Reasoner, and AUDIO-MIND on the same 1,000-example evaluation split.

The sub-category results suggest that AUDIO-MIND helps most when a question can benefit from bounded, task-specific evidence rather than generic decomposition. The largest gains over Qwen3.5-Omni appear in Music Theory (+15.87), Temporal Analysis (+14.29), Anomaly Detection (+17.65), and Correlation Analysis (+8.00), which are categories where structured measurements, temporal localization, or transformed audio segments can provide concrete support for the frontend. In contrast, AUDIO-MIND does not consistently improve categories that rely mainly on holistic semantic judgment or perceptual interpretation: it is slightly worse on Spatial Analysis, Acoustic Quality Analysis, and Audio Difference Analysis. This mixed pattern shows that there is no universally optimal division of labor between external tools and frontend LALMs. The capability boundary is question dependent: tools may provide more reliable evidence for some subcategories, while direct frontend perception may remain stronger for others. As both LALMs and specialized tools improve, this boundary may also shift across model generations and task settings. Estimating when tool evidence is likely to help, and when it may distract from stronger frontend judgment, is therefore an important direction for audio agent research.

B System Behavior and Tool-Usage Statistics

This appendix expands on Section 5.3 with full distributions over planner rounds, tool calls, tool-usage frequency, and targeted re-listening. Unless otherwise stated, all behavioral statistics are computed on the 999 successfully completed MMAR runs; one of the 1,000 evaluation examples failed due to an API content-safety rejection and is excluded from tool-use statistics.

B.1 Planner-Round Distribution

A planner round is one decision step in which the planner emits either a tool call, a re-listen request, an answer, or a give-up action. Across the 999 runs, the planner uses on average 2.68 rounds per question. 99.9% of runs exit with an ANSWER action; only one run exits with FAIL due to an API

Layer	Sub-category	N	Qwen	AudioGenie	AUDIO-MIND	Δ vs. Qwen	Δ vs. AudioGenie
Cultural	Aesthetic Evaluation	8	75.00	50.00	62.50	-12.50	+12.50
Cultural	Culture of Speaker	52	78.85	76.92	82.69	+3.85	+5.77
Cultural	Imagination	10	70.00	60.00	80.00	+10.00	+20.00
Cultural	Professional Knowledge and Reasoning	71	76.06	74.65	76.06	+0.00	+1.41
Perception	Correlation Analysis	50	80.00	74.00	88.00	+8.00	+14.00
Perception	Counting and Statistics	99	64.65	56.57	63.64	-1.01	+7.07
Perception	Environmental Perception and Reasoning	149	87.25	70.47	85.91	-1.34	+15.44
Perception	Music Theory	63	57.14	55.56	73.02	+15.87	+17.46
Perception	Spatial Analysis	15	73.33	73.33	60.00	-13.33	-13.33
Perception	Temporal Analysis	28	64.29	42.86	78.57	+14.29	+35.71
Semantic	Content Analysis	304	85.20	80.26	86.84	+1.64	+6.58
Semantic	Emotion and Intention	60	83.33	65.00	76.67	-6.67	+11.67
Semantic	Speaker Analysis	48	87.50	77.08	87.50	+0.00	+10.42
Signal	Acoustic Quality Analysis	18	77.78	55.56	66.67	-11.11	+11.11
Signal	Anomaly Detection	17	70.59	70.59	88.24	+17.65	+17.65
Signal	Audio Difference Analysis	8	62.50	50.00	37.50	-25.00	-12.50

Table 5: Full MMAR sub-category accuracy comparison. Values are accuracies in percentage points. AudioGenie denotes AudioGenie-Reasoner instantiated with Qwen3.5-Omni and Qwen3.5 for matched-backbone comparison.

Rounds	Count	Share
1	311	31.1%
2	255	25.5%
3	164	16.4%
4–5	203	20.3%
6–10	59	5.9%
>10	7	0.7%

Table 6: Distribution of planner rounds per MMAR question.

Tool	Calls	% of calls	Q-share
`transcribe_qwenasr_with_timestamps`	227	14.9	18.9
`inspect_audio_plots`	206	13.6	11.1
`trim_audio`	173	11.4	6.9
`transcribe_fireredasr_with_timestamps`	113	7.4	11.3
`analyze_pitch`	107	7.0	5.4
`analyze_onsets`	85	5.6	7.5
`transcribe_whisperx_with_diarization`	64	4.2	6.2
`extract_rms_energy`	58	3.8	3.7
`analyze_spectral_features`	55	3.6	3.8
`transcribe_qwenasr`	46	3.0	3.7

Table 7: Top-10 most invoked tools on MMAR. Q-share is the percentage of questions invoking the tool at least once.

content-safety rejection. Table 6 reports the full distribution: 31.1% of questions exit in a single round (i.e., the planner emits ANSWER immediately after the initial caption), and 93.3% exit within five rounds.

B.2 Tool-Usage Frequency

Across 999 MMAR runs, AUDIO-MIND invokes 1,675 tools in total. Table 7 lists the ten most frequently used tools, which together account for 74.5% of all invocations. The distribution is heavy-tailed: ASR with timestamps, audio-plot inspection, and time-range trimming dominate, consistent with the prevalence of speech, sound, and temporal-localization questions in MMAR. Of the remaining 25 tools each accounts for less than 3% of calls, but each is used on at least one question, indicating that the long tail of the inventory provides bounded, occasionally decisive coverage rather than redundant capacity. We also observed 6 invocations of a non-existent tool name (`content`) across 4 questions; these are planner hallucinations rejected by the executor and represent a minor failure mode in tool selection.

B.3 Targeted Re-Listening

Targeted re-listening, in which the planner sends a selected audio identifier and a focused prompt back to the frontend, is used sparingly: only 50 out of 999 MMAR questions (5.0%) trigger at least one re-listen, for 64 re-listens in total. This pattern is consistent with the intended role of re-listening: it is reserved for cases where tool-side transformation or selection has changed what the frontend can perceive, rather than serving as a generic fallback.

B.4 Long-Tail >10-Call Bucket

In Section 5.3, the > 10 tool-call bucket on MMAR contains only six questions and is too small to support a reliable interpretation. On these six questions, the direct Qwen3.5-Omni baseline achieves 50.0% accuracy while AUDIO-MIND drops to 16.7%. Possible explanations include long tool chains accumulating noisy intermediate evidence, planner instability on the hardest examples, and selection bias — only the most uncertain questions reach this depth. Distinguishing these explanations would require more examples than this benchmark provides at the tail; we therefore treat the bucket as

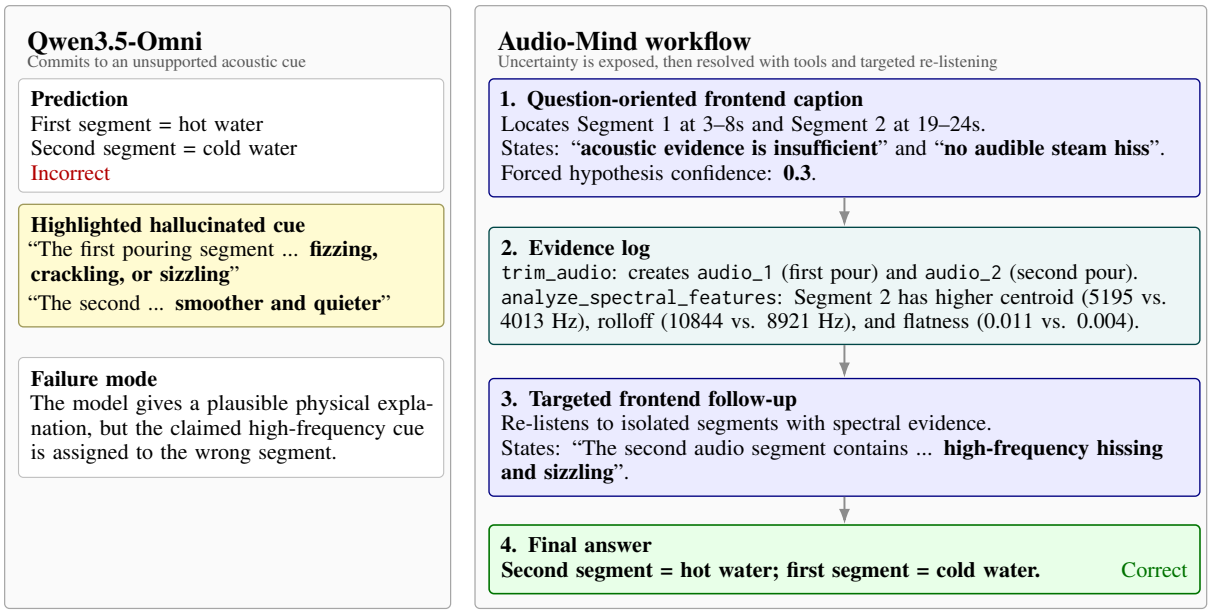

Figure 2: Qualitative correction trace for a hot-versus-cold water example. The left panel shows the Qwen3.5-Omni response, where highlighted text marks the unsupported acoustic cue that leads to the wrong answer. The right panel shows the agent path: uncertainty is stated in the initial caption, tools isolate and measure both pouring segments, and targeted frontend follow-up uses this evidence to produce the correct answer.

suggestive of a degradation regime worth investigating, but do not draw conclusions from it.

C Qualitative Case Study

We include one qualitative case to illustrate three behaviors that motivate the agent design. First, **an LALM may hallucinate a plausible acoustic rationale**: in an MMAR example asking which of two water-pouring segments is hot, Qwen3.5-Omni answers incorrectly by attributing a fizzing or sizzling cue to the first segment. Second, **the question-oriented frontend prompt helps expose uncertainty**: under our prompt, the same frontend localizes the two pours at approximately 3–8s and 19–24s, but marks the raw perceptual evidence as weak, notes that no clear steam hiss or ice clink is audible, and assigns only 0.3 confidence to its forced hypothesis. Third, **tools provide the missing evidence**: AUDIO-MIND isolates the two candidate segments, measures their spectral properties, and asks the frontend to re-listen with this comparative evidence. The second segment shows stronger high-frequency noise-like energy, with higher spectral centroid (5195 Hz vs. 4013 Hz), rolloff (10844 Hz vs. 8921 Hz), and flatness (0.011 vs. 0.004), leading targeted re-listening to reverse the initial forced option and select the correct answer. This example does not establish broad tool reliability by itself, but illustrates how AUDIO-MIND turns exposed uncertainty into focused evidence acquisition, as shown in Figure 2.

D Reproducibility Details

The MMAR configuration uses `qwen3.5-omni-plus` for the audio-capable frontend and `qwen3.5-plus` for the text planner. The MSU-Bench configuration uses `Gemini 2.5` with thinking enabled as the audio-capable frontend and `Gemini 3.0 Flash` with thinking enabled as the text planner. We use temperature 0.05 for model calls and allow at most 15 planner decision steps per example. We use all pretrained models, benchmarks, and external tools under their respective licenses or terms of use, only for research evaluation and analysis, and do not redistribute third-party model weights, benchmark data, or tool artifacts. The prompt templates, tool inventory, and evaluation scripts are included with the submitted supplementary materials and will be released with the code.

E Tool Inventory

Table 8 summarizes the tool inventory used in our experiments. We use the term "tool" broadly: some entries wrap specialized pretrained models, such as ASR, diarization, speaker verification, or chord recognition systems, while others are deterministic or library-based audio operations implemented

with standard signal-processing utilities such as FFmpeg (Tomar, 2006) and librosa (McFee et al., 2015). The planner sees both types through the same bounded interface, but their expected outputs and reliability assumptions are specified separately in the tool-inventory configuration. Detailed input schemas and boundary instructions are maintained in the planner configuration.

F Generative AI Use Disclosure

The authors used AI writing assistants to support language polishing, phrasing refinement, and checklist preparation. All scientific ideas, system design, experiments, analyses, and final manuscript decisions were made and verified by the authors.

Tool	Description
Metadata validation	
Audio stream statistics	Reports audio stream metadata and summary statistics.
Loading metadata	Reports duration, sample rate, and basic loading metadata.
Audio derivation	
FFT denoising	Creates a denoised audio file using FFT-domain filtering.
Wavelet denoising	Creates a denoised audio file using wavelet filtering.
Channel conversion	Converts audio to a target channel layout.
High-pass filtering	Attenuates low frequencies below a cutoff.
Low-pass filtering	Attenuates high frequencies above a cutoff.
Resampling	Converts audio to a target sample rate.
Harmonic–percussive separation	Estimates harmonic and percussive source components.
Time-range trimming	Cuts a selected time span into a derived clip.
Temporal segmentation	
Speech/music activity detection	Detects coarse speech, singing, and music regions.
FireRed speech VAD (Xu et al., 2026)	Detects likely speech activity regions.
Audio segmentation	Splits audio using fixed-duration or silence-based rules.
Silence detection	Detects silence intervals with threshold and duration settings.
Silero VAD	Detects likely speech regions with Silero VAD.
Speech and speaker processing	
DiariZen diarization (Han et al., 2025)	Estimates anonymous speaker-labeled time segments.
SortFormer diarization (Park et al., 2025)	Estimates speaker-labeled segments with arrival-order tracking.
Speaker verification (Wang et al., 2023)	Compares two clips for same-speaker evidence.
FireRedASR (Xu et al., 2025b, 2026)	Transcribes speech without word timestamps.
FireRedASR with timestamps (Xu et al., 2026)	Transcribes speech with rough timing information.
Qwen3-ASR (Shi et al., 2026)	Transcribes speech with the Qwen ASR model.
Qwen3-ASR with timestamps (Shi et al., 2026)	Transcribes speech with timestamp information.
WhisperX ASR (Bain et al., 2023)	Transcribes speech with WhisperX.
WhisperX ASR with diarization (Bain et al., 2023; Bredin, 2023; Plaquet and Bredin, 2023)	Produces transcription with diarization output.
Acoustic/music feature analysis	
Onset analysis	Detects onset times and onset-strength cues.
Pitch contour analysis	Estimates rough pitch contours and pitch statistics.
Tempo-CNN estimation (Schreiber and Müller, 2018)	Estimates two likely tempi and relative salience.
Spectral descriptors	Computes centroid, bandwidth, rolloff, contrast, and flatness.
Amplitude statistics	Computes detailed amplitude statistics.
Musical key estimation	Estimates a coarse musical key center.
Chroma extraction	Extracts pitch-class energy features over time.
MFCC extraction	Extracts spectral-envelope features.
RMS energy extraction	Extracts an energy-envelope proxy.
Large-vocabulary chord recognition (Jiang et al., 2019)	Recognizes chord timelines with extended chord vocabulary.
FFmpeg spectral statistics	Computes spectral statistics over audio.
Volume statistics	Computes mean and maximum volume statistics.
Signal visualization inspection	
Audio-plot inspection (Bai et al., 2025)	Inspects waveforms, spectrograms, or other audio plots for bounded signal-level cues.
Processing-quality verification	Checks whether trimmed, filtered, separated, or denoised audio remains usable for downstream analysis.

Table 8: Full planner-facing tool inventory used by AUDIO-MIND. Each category header groups tools with a shared operational role; detailed tool schemas and boundary instructions are specified in the planner inventory configuration.