# WHEN SHOULD A SPOKEN AGENT ACT? A BENCHMARK FOR RISK-AWARE FUNCTION CALLING

**Anonymous authors**
Paper under double-blind review

## ABSTRACT

Spoken agents can send messages, control devices, and initiate payments from voice commands. Automatic speech recognition (ASR) uncertainty about a recipient or amount therefore creates an authorization problem: should the agent act, ask one question, or stop? Word error rate and function-call accuracy do not resolve that decision. We introduce Risk-Aware Spoken Function Calling (RASFC), a controlled rendered-speech benchmark that pairs structured calls with risk tiers, argument-support labels, and rule-derived next actions: execute, ask a targeted clarification, or abstain. The current benchmark interface gives its verifier the intended action's risk tier at evaluation time; this oracle context is distinct from risk inferred from speech or a predicted call. In a replayable offline diagnostic on 6,000 distinct audio/ASR observations, reference-WER plus ASR confidence misses 1,147 dialogue-required examples under a rule built from gold argument support, clarification targets, risk, and confidence, whereas a gold argument-support diagnostic misses 389. These signal comparisons measure consistency with the constructed rule. RASFC makes the information flow explicit for evaluating when uncertain speech warrants action, targeted repair, or abstention; it is a controlled benchmark interface, not an end-to-end human-speech deployment evaluation.

## 1 INTRODUCTION

Voice interfaces increasingly send confidential messages, route users, control devices, and initiate payments. In these settings, a transcription error can change the action itself. A recognizer may assign similar support to "send fifty dollars to Alice", "send fifteen dollars to Alice", and "send fifty dollars to Alex", although the safe response differs in each case. The assistant must decide whether the available evidence supports execution, identifies one argument to clarify, or is too uncertain to act. We ask how a spoken agent should make that decision.

Standard recognition and parsing metrics do not answer this question. Word error rate (WER), exact-call accuracy, and slot F1 measure transcript or semantic correctness, but not whether a wrong call is executed or whether a focused question could repair it. In an offline diagnostic that computes WER from clean reference text, 2,613 of 3,664 low-WER utterances still require clarification or abstention; among low-reference-WER examples with high ASR confidence, 1,147 of 1,751 still do. The reported scalar confidence score reaches only 0.551 area under the receiver operating characteristic curve (AUROC) for ranking unsupported arguments. Across matched renderings of the same command, 353 of 3,000 source groups change their inferred action, including 286 execution-versus-dialogue conflicts. These results describe disagreement between transcript-quality signals and a constructed dialogue rule. The rule depends on gold argument support and confidence, so agreement with it cannot independently validate actionability.

We study their intersection through *risk-aware spoken function calling*. The system predicts both a structured call and a next action: *execute* when every critical argument is supported, *clarify* when one recoverable argument is uncertain, and *abstain* when the ambiguity is diffuse. RASFC[1] makes this distinction testable by pairing function calls with risk tiers, argument-support labels, and valid clarification targets.

[1] Anonymous source code: `https://anonymous.4open.science/r/iclr27-rasfc-3B80/`



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

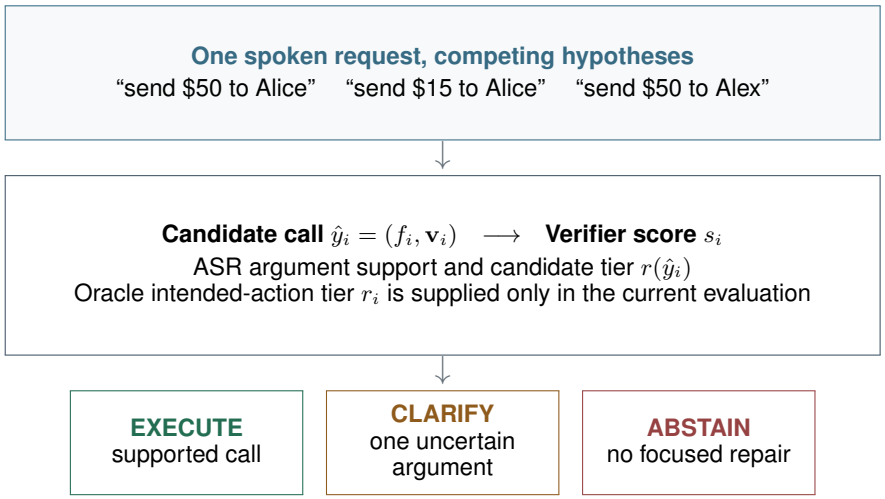



Figure 1: Risk-aware spoken function calling. Competing speech hypotheses provide different support for a candidate call. Candidate-schema metadata $r(\hat{y}_i)$ and the current evaluation-only oracle tier $r_i$ are distinct. The policy executes only a supported call, asks about one recoverable argument, or abstains when the evidence remains insufficient.

**Contributions.**

1. We formulate a three-action authorization evaluation for spoken function calls and measure false execution, high-risk execution coverage, targeted clarification, and utility jointly.
2. We provide the RASFC benchmark, which separates structured-call correctness, candidate-schema metadata, the oracle intended-action risk tier used by the current evaluation, argument support, and valid clarification targets.
3. We provide replay-aware baseline diagnostics that distinguish recognizer-only signals, reference/oracle rule-consistency measurements, and direct-audio comparators, while keeping earlier summaries without saved predictions out of ranked method comparisons.

Figure 1 summarizes how uncertain speech becomes structured call evidence and then a risk-aware execution decision.

## 2 Related Work and Gap

**Spoken function calling.** Spoken Function Calling proposes function-call prediction as a spoken language understanding target for large audio language models Peng et al. (2026b). Audio2Tool benchmarks speech tool use across domains, acoustic conditions, and compositional difficulty Pahwa et al. (2026). The SLT SmartGlasses Challenge pushes spoken understanding toward egocentric multi-talker wearable audio Gao et al. (2026). Our extension changes the decision problem: the model must decide whether a predicted call is safe enough to execute and, when it is not, whether a targeted clarification can resolve the uncertainty. This adds risk-tiered false execution, selective utility, and recoverable-argument evaluation. RASFC therefore adds an execution-authorization decision to spoken actions.

**Reliable audio-language models.** Large audio language model (LALM) reliability has been studied through Reliable LALM, Audio-CoT, AnyAudio-Judge, efficient chain-of-modality reasoning, SLAM-Omni, AHAMask, VISA, and SpeechAgent-R, which address instruction following, reasoning, evaluation, modality compression, robust task specification, evidence-grounded audio reasoning, and tool-interactive audio agents Ma et al. (2025b;a); Li et al. (2026a); Feng et al. (2026); Chen et al. (2025); Guo et al. (2026); Tu et al. (2026); Wang et al. (2026). Reliable LALM is the closest refusal anchor, and AHAMask is the clearest task-specification anchor. Our setting begins after those issues: even with a correctly specified task, the agent must decide whether uncertain spoken arguments authorize a risky action. These works motivate stronger audio agents; our evaluation adds execution authorization under ASR uncertainty and the operational cost of a wrong structured action.

**Interactive ASR and semantic correction.** Interactive ASR, AgenticASR, proactive uncertainty clarification, CTC-assisted contextual ASR, MaLa-ASR, and READ use semantic feedback, error diagnosis, clarification, multimodal context, or acoustic discrepancy to improve or evaluate recognition and correction Jiang et al. (2026b;a); Peng et al. (2026a); Yang et al. (2024a;b); Li et al. (2026b). Those systems ask what transcript correction or follow-up interaction best improves recognition. Although they improve recognition, candidate uncertainty can still change whether a risk-critical argument is supported and whether high-risk false execution occurs. RASFC is complementary: it asks a downstream authorization question over the current spoken evidence. Even an improved transcript may still be unsafe to execute if the recipient, amount, time, or destination remains weakly supported, while a slightly imperfect transcript can already justify the correct action. We therefore make ASR uncertainty the dialogue state that governs executable tool use, treat uncertainty at the argument level, and reward clarifications that resolve the specific unsupported argument.

**Speech generation and deployment systems.** X-Translator, X-Voice, GROW, and Semantic-VAE study real-time translation, multilingual voice cloning, and speech synthesis Zhao et al. (2026); Xu et al. (2026); Yang et al. (2026); Niu et al. (2026). We use rendered speech as a controlled stress medium, not as evidence about human deployment.

Across these lines of work, existing benchmarks predict calls, improve transcripts, or study broad refusal behavior. RASFC adds an authorization decision tied to argument support and evaluates it with high-risk false execution, execution coverage, utility, and clarification precision. The contribution is therefore a task-and-evaluation interface for deciding whether uncertain speech supports action, a focused repair, or abstention.

**Selective execution.** Selective prediction motivates abstention, while spoken tool use also needs targeted clarification. A safe assistant asks about the minimal ambiguous argument when recovery is possible and abstains when recovery is unavailable. Our current implementation uses development-selected tier-specific thresholds; it does not assert a conformal coverage guarantee. Clarification precision and high-risk false execution are therefore first-class metrics.

## 3 Task and Benchmark

Each example contains a spoken command, a gold function call, a risk tier, perturbation labels, and a valid clarification target. The caller receives the ASR observation, schema inventory, legal candidates, and candidate-derived support at test time. The verifier and threshold policy additionally receive $r_i$, the benchmark-assigned risk tier of the intended action. This is oracle benchmark context, not a risk value inferred from speech or the selected call. Table 1 separates these inputs from reference annotations that remain outside current test inference. The current supervised direct-audio support-decoder code also uses benchmark perturbation indicators at test time; the earlier reported decoder results have not been checked against that code and cannot verify an annotation-free information boundary. The system predicts a structured call and a decision. Execution is rewarded only when the call is correct; wrong execution is penalized most strongly for high-risk examples. Clarification receives partial credit when it asks about the right recoverably uncertain argument; abstention receives a small penalty.

Formally, for utterance $x_i$ with gold call $y_i$ and risk tier $r_i$, a system outputs $(\hat{y}_i, d_i, q_i)$, where $d_i \in \{\text{execute}, \text{clarify}, \text{abstain}\}$ and $q_i$ is an optional clarification target. Let $\mathcal{H} = \{\text{high}\}$ denote the high-risk tier, and let $I[\cdot]$ be the indicator function. We report high-risk false execution among executed high-risk examples,

$$\text{HRFE} = \frac{\sum_i I[r_i \in \mathcal{H}, d_i = \text{execute}, \hat{y}_i \neq y_i]}{\sum_i I[r_i \in \mathcal{H}, d_i = \text{execute}]},$$

When no high-risk example is executed, HRFE is not estimable; we report this state with its execution numerator and denominator instead of interpreting a zero-denominator convention as zero error. The benchmark utility is asymmetric: correct execution is $+1$, non-high-risk false execution is $-2$, high-risk false execution is $-4$, targeted clarification is $+0.35$, untargeted useful clarification is $+0.05$, spurious clarification is $-0.25$, and abstention is $-0.15$ for high-risk or $-0.5$ otherwise. These are benchmark costs, not universal deployment costs.

This objective differs from ordinary ASR correction because the hidden variable includes executable support, not just a word sequence. Let $z_i = (y_i, m_i)$ denote the executable dialogue state, where $m_i$

Table 1: Information supplied to the current evaluation paths. Candidate schema tier $r(\hat{y}_i)$ is fixed metadata of the selected legal candidate. Intended-action tier $r_i$ is oracle benchmark context, not information inferred from speech or the selected call.

| Information | Source | Current test use |
|---|---|---|
| ASR transcript, word confidence, and word probabilities | Recognizer output | Caller and verifier |
| Legal schemas, values, aliases, and $r(\hat{y}_i)$ | Fixed schema inventory and selected candidate | Caller and verifier |
| Candidate and argument support | ASR observation plus legal candidates | Caller and verifier |
| Intended-action tier $r_i$ | Gold benchmark annotation | Verifier and tier-specific thresholding (oracle) |
| Perturbation indicators | Benchmark annotations | Current support-decoder test inputs; report-to-code match unverified |
| Clean text, gold call, and clarification target | Supervision and evaluation annotations | Not current caller inputs |

records which risk-critical arguments are sufficiently supported for action. ASR hypotheses, word confidences, acoustic evidence, and candidate calls are observations of $z_i$. The policy is evaluated by the posterior action it induces: execute when posterior mass concentrates on one fully supported call, clarify when posterior uncertainty localizes to one recoverable argument, and abstain when uncertainty remains diffuse. Two transcripts can be equally close to the reference while implying different safe actions, which makes WER insufficient for this decision.

The replayable dialogue diagnostic uses 3,000 source commands, each with one Apple and one VITS audio/ASR observation (6,000 observations, balanced across risk tiers). Each observation is counted once. Example-level counts refer to observations; cross-renderer action counts refer to commands. Source-disjoint policy selection uses 2,075 training commands and 925 held-out commands from this same cohort. The 5,000-example prototype, 10,000 tempo examples, 9,000 MMS-TTS examples, older matched-risk caller summaries, and direct-audio studies use separate protocols. Their source overlap is unverified, so we neither pool them nor treat them as independent replications. Appendices A and C detail the study units and protocols.

RASFC also separates three labels that are often conflated in spoken-agent evaluation. The function-call label measures semantic extraction, the risk tier measures the consequence of executing the call, and the clarification target identifies the smallest argument that should be repaired before action. This separation lets a system be rewarded for a useful follow-up question without pretending that the first-turn call was correct. It also makes failure analysis operational: an error on a payment amount, a message recipient, or a destination is counted as an execution-policy failure, not only as a slot mismatch.

## 4 METHOD

Within this benchmark, a point transcript discards competing argument support needed for the risk-aware action choice. We therefore retain candidate-level support and uncertainty before selecting the action.

We represent each decision with a dialogue belief state containing a candidate posterior, an argument-support vector, and an unsupported-slot set. Candidate schema tier $r(\hat{y}_i)$ is fixed metadata of the selected legal candidate. The current verifier and tier-specific thresholding additionally receive oracle intended-action tier $r_i$. The policy maps this state to execute / clarify / abstain: it executes only when one fully supported call dominates, clarifies a recoverable unsupported slot, and abstains when the posterior remains diffuse. This makes RASFC a Bayes-risk benchmark action problem with asymmetric execution and clarification costs. It is not a blind speech-only policy evaluation.

**Constrained candidates.** We enumerate legal schema candidates from function templates and allowed argument values. A deterministic constrained caller scores candidates using lexical overlap and argument support, exposing candidate score, margin, top probability, and entropy.

**Risk verifier.** A lightweight verifier scores selected-call risk using oracle intended-action tier $r_i$, base risk score, argument coverage, ASR confidence, word-level argument probabilities, constrained-candidate uncertainty, noisy-channel support, and direct-audio agreement. We denote its scalar output by $s_i \in [0, 1]$; lower values are safer within this benchmark. Development thresholds satisfy $\tau_{\text{exec}} < \tau_{\text{abs}}$: the policy executes below the first threshold when support floors pass, clarifies in the interval when a recoverable slot exists, and abstains otherwise. The targeter is fit on training clarification annotations and predicts a target from selected-candidate support features and schema metadata at development and test time. Its output is scored by the clarification metric and is not an input feature to the verifier. Thresholds are selected on development data; test HRFE is an empirical benchmark quantity, not a deployment guarantee under distribution shift.

**Noisy-channel caller.** The main model learns how clean schema values are corrupted by ASR. The paired clean transcript is used only as training supervision and is never supplied to the caller or policy at inference. At test time, the caller uses the ASR observation, schema inventory, spoken aliases, lexical support, and argument-level channel scores. The caller is designed to distinguish support for names, numbers, times, destinations, and device states more precisely than deterministic template matching.

**Risk control.** We select tier-specific decision thresholds on development splits. We report the plus-one high-risk execution diagnostic $(e+1)/(n+1)$ on the held-out benchmark, where $e$ is the number of high-risk false executions and $n$ is the number of executed high-risk examples. This diagnostic is not a distribution-free or deployment guarantee.

**Direct-audio baselines.** We evaluate prompt-only Ultravox candidate choosers using JSON-only choices, rendered natural wording, risk-aware prompts, two-pass confirmation, and checklist/few-shot prompts. We also train a metadata-only candidate-support classifier and a small support head on frozen Whisper waveform embeddings. A pooled variant trains the same head across Apple, VITS-LJS, and eSpeak renderings, with its high-risk support floor selected on development data. Prompt-only comparators receive waveform input and the legal candidate inventory. The current trained support-decoder implementation also receives annotation-derived perturbation indicators; the saved result summaries have not been matched to that implementation, so they cannot isolate waveform-only support. Clean reference transcripts are used as training supervision for the paired noisy-channel caller.

The pipeline is modular so that ASR, constrained candidate generation, verifier scoring, tier-specific threshold selection, clarification targeting, and direct-audio comparison remain separate checked outputs. This design is useful for a benchmark paper because each decision can be audited, swapped, or stress-tested without changing the task definition. In particular, the input audit separates better transcription, schema matching, argument support, candidate metadata, and oracle benchmark context.

## 5 EXPERIMENTS

We evaluate Whisper-family ASR backends and faster-whisper large-v3 outputs on controlled rendered-speech conditions. All thresholds are selected on development data and applied once to their test split. We distinguish replayable diagnostics with retained per-example traces from older summaries that cannot be tied to their original split or model.

Metrics are exact-call accuracy, slot F1, execute rate, false execution, high-risk false execution (HRFE) when its denominator is nonzero, clarification precision, and label-based targeted clarification precision, together with utility. The clarification precision measures are annotation-based repair diagnostics, not measures of human clarification naturalness. Thresholds are selected on development splits and applied once to test splits.

The experiments answer three questions. First, do reference/recognizer diagnostics identify the benchmark's annotated execute / clarify / abstain decision? Second, does argument-support information resolve ambiguity left by those diagnostics under the oracle-risk benchmark interface? Third, does direct waveform access provide enough candidate-specific evidence for this diagnostic? We report positive and negative results because a policy with no executed high-risk examples does not identify a high-risk error rate, and a useful benchmark diagnostic is not a deployment evaluation.

Table 2: Offline reference/recognizer equivalence classes are not dialogue-equivalence classes. Reference WER uses clean text; task conditioning includes the oracle risk tier; gold argument support enters the final profile. Exec./dialogue conflicts are entries in classes containing both executable and non-executable actions.

| Signature | Buckets | Mixed ↓ | Exec./dlg. ↓ | Low-WER/conf. ↓ |
|---|---|---|---|---|
| Reference WER + recognizer | 11 | 6000 | 5934 | 1751 |
| Task + oracle tier + ref./recognizer | 61 | 5979 | 3974 | 1556 |
| + gold support | 216 | 2483 | 0 | 0 |

Adding gold argument-support state eliminates execute/clarify ambiguity under the same task, oracle-tier, reference-WER, and recognizer bins in this diagnostic.

# 6 RESULTS

The execution fractions reported on the matched-risk split are split-level diagnostics for the supplied benchmark, not estimates of population deployment coverage; any deployment coverage statement requires an external operating distribution. All risk statements are conditional on the oracle benchmark risk labels and selected cost specification.

Figure 2 isolates the central failure mode with replayable offline diagnostics. Reference WER plus ASR confidence misses 1,147 dialogue-required turns, including 364 high-risk turns; reference WER uses clean text and is not a deployed-policy input. The gold argument-support-plus-target diagnostic misses 389 dialogue-required turns, all high-risk, but it uses gold call arguments and annotation. These reference and oracle diagnostics measure consistency with the constructed target; they do not provide independent action-label validation. The diagnostic target is true when a gold clarification target or unsupported gold argument is present, or when low confidence coincides with high or critical oracle risk. Appendix B defines the rule and its inputs. An information audit measures agreement with this target: reference-WER, recognizer-confidence, and mixed buckets explain at most 0.048 normalized mutual information with the execute/clarify/abstain action, versus 0.703 for oracle argument support and 0.798 for the full oracle-risk–support–confidence diagnostic.

Table 2 provides a stricter equivalence audit. After conditioning on task type, the oracle risk tier, reference-WER/recognizer buckets, 3,974 cases still mix executable and non-executable actions, while adding the oracle supported-argument set removes execute/dialogue conflicts in this constructed diagnostic. A threshold audit makes the negative control operational: even oracle WER thresholds recover zero true executions at zero false execution, and the utterance-log-probability gate recovers only two true executions at zero high-risk false execution. The ASR candidate stream exposes the same ambiguity before reranking: 300/8,354 waypoint-bearing examples have competing transcripts that disagree on whether a risk-critical waypoint is present. A source-held-out majority policy reaches 0.321 action accuracy from the best recognizer-only bucket and 0.826 from the oracle belief proxy, with no high-risk false executions. Because the target itself uses argument support, risk, and confidence, the oracle proxy partly reconstructs the target rule. The source split prevents command overlap but does not remove this dependence. These are internal-consistency diagnostics, with no independently measured authorization benefit.

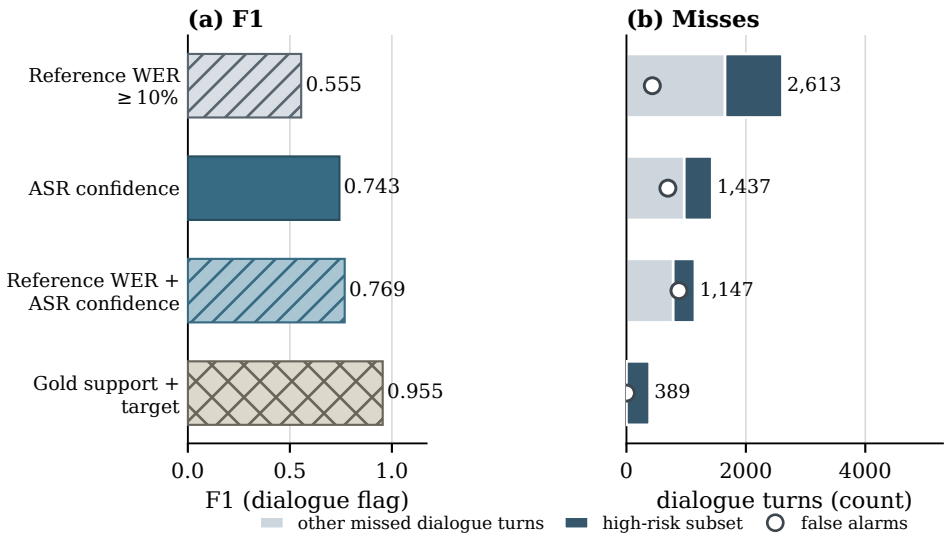
## 6.1 CALLER AND REPAIR RESULTS WITH MISSING REPLAY INPUTS

Appendix B.1 gives these results; absent traces preclude a paired ranking.

Cross-renderer analysis finds 353 of 3,000 source groups whose annotated next action changes across ASR observations, including 286 execution-versus-dialogue conflicts and 163 high-risk groups. This measures sensitivity of the benchmark labels to speech observations; it is not a comparison of the earlier reported noisy-channel result with the candidate-rescoring diagnostic.

## 6.2 ACOUSTIC REALISM AND DIRECT AUDIO

Table 3 summarizes the stress tests. Apple platform text-to-speech and a VITS model trained on LJSpeech (VITS-LJS) show that lower WER and better exact-call accuracy still do not imply low observed benchmark error. The VITS-LJS split has low WER (8.0%) but direct execution has 58




n=6,000; 4,516 dialogue-required; 1,781 high-risk dialogue-required.

Figure 2: Replayable offline diagnostic on 6,000 distinct audio/ASR observations from 3,000 source commands, each observed with Apple and VITS speech. Left: F1 for matching dialogue-required turns computed from gold targets, argument support, risk, and confidence. Right: total misses, with the high-risk subset shaded darker and false alarms marked separately. Reference WER uses clean reference text; gold support plus target uses gold call arguments and annotation. Oracle features overlap with the target construction, so this figure measures internal consistency.

Table 3: Acoustic-realism and direct-audio stress tests. Direct uses the ASR transcript without channel correction; noisy-channel fits channel evidence on the same renderer; transfer reuses the channel learned on the matched eSpeak benchmark. Strict utilities use development-selected tier-specific thresholds with oracle benchmark risk tiers. Strict high-risk execution coverage is shown explicitly; strict HRFE is not estimable for these zero-coverage points.

| **Condition** | **WER (%) ↓** | **Exact ↑** | **Strict HR exec./avail.** | **Strict util. ↑** |
|---|---|---|---|---|
| Apple 3k direct | 18.0% | – | 0/150 | 0.163 |
| Apple 3k noisy-channel | 18.0% | 0.807 | 0/150 | 0.369 |
| Apple 3k transfer | 18.0% | 0.742 | 0/150 | 0.386 |
| VITS-LJS 3k direct | 8.0% | – | 0/150 | 0.067 |
| VITS-LJS 3k transfer | 8.0% | 0.780 | 0/139 | 0.335 |
| VITS-LJS rendered Ultravox | 8.0% | 0.433 | 0/155 | 0.095 |

high-risk false executions among 244 high-risk executions. Every listed strict policy executes zero high-risk cases, so its strict HRFE is not estimable; its utility is a separate oracle-risk benchmark diagnostic.

Prompt-only direct-audio baselines remain weak: VITS-LJS rendered Ultravox reaches 0.095 strict utility, two-pass confirmation reaches 0.085, and risk-confirmed prompting reaches 0.073. A separate 9k eSpeak-rerendered self-attention run remains a negative control: raw utility is -0.239 with 13 high-risk false executions among 59 high-risk executions, and its development-selected test policy executes only 4.9% of examples with -0.015 utility without identifying a high-risk error rate. Checklist and few-shot high-risk smoke tests execute no high-risk examples. Prompt-side waveform access and lower WER do not by themselves provide calibrated candidate-specific support. A threshold-sensitivity audit separates points with no executed high-risk examples from waveform-decoder points with nonzero high-risk execution counts, keeping the oracle-risk benchmark statement tied to its denominator.

Table 4: Direct-audio prompt diagnostics before tier-specific thresholding. Exec. is execute rate; util. is raw selective utility. HR false/exec. reports observed high-risk false executions over high-risk executions; – denotes no high-risk execution.

| Prompt variant | n | Exact ↑ | Exec. | HR false/exec. | Util. ↑ |
|---|---|---|---|---|---|
| Apple risk-rendered | 3000 | 0.381 | 0.006 | 0/2 | -0.332 |
| Apple rendered yes/no | 3000 | 0.378 | 0.042 | 6/41 | -0.175 |
| Apple risk-confirmed | 3000 | 0.381 | 0.006 | 0/2 | -0.338 |
| Apple few-shot high-risk | 60 | 0.333 | 0.000 | – | -0.153 |
| VITS rendered | 3000 | 0.433 | 0.305 | 66/236 | -0.120 |
| VITS risk-confirmed | 3000 | 0.433 | 0.030 | 1/5 | -0.290 |
| eSpeak rerender self-attn | 9000 | 0.415 | 0.068 | 13/59 | -0.239 |
| VITS few-shot high-risk | 60 | 0.350 | 0.000 | – | -0.160 |

Table 4 shows why prompt-only direct audio is a controlled negative finding. Conservative Apple prompts reach 0/2 high-risk false/execute only by executing 18 of 3,000 examples, and rendered yes/no support still has negative raw utility. Trainable support models achieve higher strict utility under the oracle-risk interface. The metadata-only candidate-support classifier reaches 0.141, with zero false executions among 13 high-risk executions out of 465 eligible high-risk examples. The frozen-Whisper waveform support head reaches 0.271, with zero among 39 executions out of 155 eligible high-risk examples on Apple. Its development-selected large-v3 all-renderer variant reaches 0.335, with zero among 111 executions out of 465 eligible high-risk examples. These are separate evaluated populations, not paired gains.

The large-v3 test has 1,350 candidate-presentation rows but only 900 distinct source/audio keys: 450 VITS waveforms each occur in two candidate modes. Its 111/465 high-risk coverage counts rows, not independent recordings; none establishes human-speech performance.

The direct-audio limitation is prompt-side access: the interface does not expose candidate-specific argument support for selective execution. The support-head summaries use different evaluated populations and unresolved annotation-input bindings, so they do not isolate the effect of stronger checkpoints or support scoring.

## 7 Discussion and Limitations

**What the results establish.** First, low reference-WER and high-confidence observations can still lack transcript support for a gold argument. Second, oracle argument-support diagnostics closely match the constructed dialogue target, whose definition uses those signals. This agreement does not independently validate the target or show an authorization benefit. Third, low observed high-risk false execution must be interpreted with the number of executed high-risk examples. The current evidence does not establish a ranking between the unreplayable noisy-channel summary and separate candidate-rescoring diagnostics.

**Evaluation scope.** All reported outcomes use rendered audio. Apple and VITS-LJS text-to-speech (TTS) conditions broaden renderer diversity, but they do not establish performance on human speech. The current verifier and tier-specific threshold policy receive the oracle intended-action tier $r_i$; candidate-derived and candidate-set-conservative risk policies have not been evaluated. Clarification is evaluated against template targets, not by whether a person finds the question natural or useful. Recordings from people and user-rated clarifications remain future work; human validation of risk and clarification labels is not part of the current evidence package. The results therefore characterize a controlled rendered-speech oracle-risk benchmark, not a deployed assistant.

**Statistical and model scope.** Prompt-only Ultravox is a lower-bound direct-audio comparator, and the stronger supervised waveform models still require human-speech validation. Source-group intervals are available for direct-audio comparisons, but the saved per-instance predictions, fitted models, and original split needed for source-group resampling of the noisy-channel summary are unavailable. Replaying a retained subset does not recover those predictions, the fitted verifier, or the original split.

**Reading error together with coverage.** Table 3 shows why a small high-risk error numerator is insufficient on its own. Direct execution on VITS-LJS has 58 false executions among 244 high-risk executions, whereas every listed strict policy executes no high-risk cases. The latter outcome has an

undefined high-risk false-execution rate, not demonstrated safety. Table 4 shows a different version of the same tradeoff: a conservative Apple prompt has zero false executions among only two high-risk executions and acts on 18 of 3,000 examples overall. Reporting the execution denominator, coverage, and utility beside observed errors distinguishes selective action from near-total refusal. It also prevents a policy that avoids all consequential actions from looking more reliable than a policy that actually makes them.

**What the signal diagnostic identifies.** Figure 2 compares information sources against a dialogue-required target constructed from gold clarification targets, argument support, oracle risk, and confidence. The gold-support-plus-target diagnostic reaches 0.955 F1 but still misses 389 dialogue-required observations; those misses follow the confidence-and-risk part of the rule. Its advantage over WER-based diagnostics locates information absent from a transcript-error threshold. It does not show that a deployed model can recover gold support from speech, or that the constructed target matches a person's authorization preference. The policy results remain conditional on the supplied intended-action risk tier. Evaluating candidate-derived risk would change the information available to threshold selection and must be measured separately.

**Units and paired comparisons.** The principal replayable diagnostic has two observations for each of 3,000 source commands. Its example-level counts therefore describe 6,000 audio/ASR observations, while cross-renderer changes use the command as the unit. Other direct-audio runs and the eSpeak rerender use separate candidate streams and test populations. The large-v3 support-decoder test has 1,350 candidate presentations but 900 distinct source/audio keys because some VITS waveforms appear in two candidate modes. Treating those presentations as independent recordings would understate dependence. A matched method comparison would need retained predictions, common source groups, candidate inventories, and threshold-selection splits. Those bindings are unavailable for the earlier reported noisy-channel result, so its utility cannot be ranked against the later support-decoder diagnostics.

## 8 Ethics and Release Package

RASFC examines authorization errors that could harm users if deployed without testing. It uses fictional function schemas and rendered speech, with no recordings from people. The supplementary MIT-licensed source archive provides a deterministic synthetic smoke test; it does not reproduce the empirical summaries, for which the audio, trained weights, and per-example predictions are not distributed. Any future release of recordings from people would require explicit consent and a suitable data statement.

## 9 Conclusion

RASFC adds a benchmarked action-selection interface alongside call prediction. It makes three outcomes explicit: execute a supported call, ask about one recoverable argument, or abstain. Within the controlled rendered-speech oracle-risk setting, the replayed diagnostics measure agreement with a constructed dialogue rule. Separate waveform summaries cannot establish a matched authorization benefit. A deployable evaluation must replace the oracle tier, retain complete per-instance traces, use common candidate streams and partitions, and report false execution, coverage, and repair quality together before interpreting a low error rate as safe behavior. The immediate use of this benchmark is controlled comparison under fixed candidate inventories and action costs. Reporting the unit of analysis and the executed-case denominator prevents gains in call accuracy from being mistaken for authorization quality. Real-world evaluation also needs independent judgments of ambiguous speech and a risk estimate available at test time.

## AI USE STATEMENT

AI tools helped improve the paper's wording and readability and locate relevant work for the literature review. The paper describes these uses.

## REPRODUCIBILITY STATEMENT

Sections 3–5 describe the benchmark, evaluation inputs, splits, and metrics. Appendix A and Appendix C specify the study units and protocols; Appendix E gives the constrained caller details. The anonymous MIT-licensed source archive provides a deterministic synthetic smoke test, but not the audio, trained weights, or per-example predictions needed to reproduce the empirical summaries.

## REFERENCES

Wenxi Chen, Ziyang Ma, Ruiqi Yan, Yuzhe Liang, Xiquan Li, Ruiyang Xu, Zhikang Niu, Yanqiao Zhu, Yifan Yang, Zhanxun Liu, Kai Yu, Yuxuan Hu, Jinyu Li, Yan Lu, Shujie Liu, and Xie Chen. SLAM-omni: Timbre-controllable voice interaction system with single-stage training. In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 2262–2282. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.findings-acl.115. URL https://aclanthology.org/2025.findings-acl.115/.

Pengchao Feng, Chao-Hong Tan, Qian Chen, Wen Wang, Xiangang Li, and Xie Chen. Efficient chain-of-modality reasoning via progressive compression for spoken language models, 2026. URL https://arxiv.org/abs/2607.19932.

Dehui Gao, Zhixian Zhao, Zhennan Lin, Yujie Liao, Yuhang Dai, Yike Zhu, Longshuai Xiao, Hui Bu, Xin Xu, Xie Chen, Shuai Wang, Liumeng Xue, Zhonghua Fu, Jun Du, Eng-Siong Chng, Jun Zhou, and Lei Xie. The SLT 2026 SmartGlasses challenge: Benchmarking egocentric multi-talker speech recognition and understanding with audio-language models, 2026. URL https://arxiv.org/abs/2608.12034.

Yiwei Guo, Bohan Li, Hankun Wang, Zhihan Li, Shuai Wang, Xie Chen, and Kai Yu. AHAMask: Reliable task specification for large audio language models without instructions. In *Fortieth AAAI Conference on Artificial Intelligence*, pp. 21513–21521. AAAI Press, 2026. doi: 10.1609/AAAI.V40I26.39299. URL https://doi.org/10.1609/aaai.v40i26.39299.

Zixuan Jiang, Binghao Qiang, Jiaying Chi, Yanqiao Zhu, Kai Yu, and Xie Chen. AgenticASR: Refining speech recognition in real-world scenarios via an agentic approach, 2026a. URL https://arxiv.org/abs/2607.28175.

Zixuan Jiang, Yanqiao Zhu, Peng Wang, Qinyuan Chen, Xinjian Zhao, Xipeng Qiu, Wupeng Wang, Zhifu Gao, Xiangang Li, Kai Yu, and Xie Chen. Towards human-like interactive speech recognition with agentic correction and semantic evaluation, 2026b. URL https://arxiv.org/abs/2605.29430.

Haitao Li, Tian Tan, Yuguang Yang, Shan Yang, and Xie Chen. AnyAudio-Judge: A dynamic rubric-based benchmark and evaluator for audio instruction following, 2026a. URL https://arxiv.org/abs/2606.03116.

Zhihan Li, Hankun Wang, Yiwei Guo, Bohan Li, Xie Chen, and Kai Yu. Read what you hear: Reference-free hypotheses evaluation with acoustic discrepancy, 2026b. URL https://arxiv.org/abs/2606.04680.

Ziyang Ma, Zhuo Chen, Yuping Wang, Eng Siong Chng, and Xie Chen. Audio-cot: Exploring chain-of-thought reasoning in large audio language model. In *IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 1–6. IEEE, 2025a. doi: 10.1109/ASRU65441.2025.11434628. URL https://doi.org/10.1109/ASRU65441.2025.11434628.

Ziyang Ma, Xiquan Li, Yakun Song, Wenxi Chen, Chenpeng Du, Jian Wu, Yuanzhe Chen, Zhuo Chen, Yuping Wang, Yuxuan Wang, and Xie Chen. Towards reliable large audio language model.

In *Findings of the Association for Computational Linguistics: ACL 2025*, pp. 1000–1014. Association for Computational Linguistics, 2025b. doi: 10.18653/v1/2025.findings-acl.56. URL https://aclanthology.org/2025.findings-acl.56/.

Zhikang Niu, Shujie Hu, Jeongsoo Choi, Yushen Chen, Peining Chen, Pengcheng Zhu, Yunting Yang, Bowen Zhang, Jian Zhao, Chunhui Wang, and Xie Chen. Semantic-VAE: Semantic-alignment latent representation for better speech synthesis, 2026. URL https://arxiv.org/abs/2509.22167.

Ramit Pahwa, Apoorva Beedu, Parivesh Priye, Rutu Gandhi, Saloni Takawale, Aruna Baijal, and Zengli Yang. Audio2tool: Speak, call, act—a dataset for benchmarking speech tool use, 2026. URL https://arxiv.org/abs/2604.22821.

Yizhou Peng, Ziyang Ma, Changsong Liu, Yi-Wen Chao, Xie Chen, and Eng Siong Chng. Proactive for uncertainty: Cause-aware error diagnosis and interactive clarification for spoken dialogue systems, 2026a. URL https://arxiv.org/abs/2605.25404.

Yuezhang Peng, Yuxin Liu, Changfeng Gao, Zhifu Gao, Xiangang Li, and Xie Chen. Spoken function calling: A new perspective on spoken language understanding for large audio language models, 2026b. URL https://arxiv.org/abs/2608.05126. ACM Multimedia 2026.

Wenming Tu, Jian Gao, Yanru Huo, Yixuan Wang, Jing Peng, Bohan Li, Ziyang Ma, Tao Liu, Shuai Fan, Kai Yu, Xie Chen, and Zilong Zheng. VISA: A visual information strengthened audio-reasoning system for the interspeech 2026 ARC agent track, 2026. URL https://arxiv.org/abs/2606.07264.

Yuwen Wang, Tian-Hao Zhang, Minghao Cai, Yilin Ren, Ziyang Jiang, Xin Wang, Zhichao Wang, Pan Zhou, Kun Zhan, and Xinyuan Qian. Hear, invoke, and understand: A skill-calling multimodal agent for large audio language models, 2026. URL https://arxiv.org/abs/2608.01881.

Rixi Xu, Qingyu Liu, Haitao Li, Yushen Chen, Zhikang Niu, Yunting Yang, Jian Zhao, Ke Li, Berrak Sisman, Qinyuan Cheng, Xipeng Qiu, Kai Yu, and Xie Chen. X-voice: Enabling everyone to speak 30 languages via zero-shot cross-lingual voice cloning, 2026. URL https://arxiv.org/abs/2605.05611.

Guanrou Yang, Ziyang Ma, Zhifu Gao, Shiliang Zhang, and Xie Chen. Ctc-assisted llm-based contextual ASR. In *IEEE Spoken Language Technology Workshop (SLT)*, pp. 126–131. IEEE, 2024a. doi: 10.1109/SLT61566.2024.10832154. URL https://doi.org/10.1109/SLT61566.2024.10832154.

Guanrou Yang, Ziyang Ma, Fan Yu, Zhifu Gao, Shiliang Zhang, and Xie Chen. MaLa-ASR: Multimedia-assisted LLM-based ASR. In *Interspeech*, 2024b. URL https://arxiv.org/abs/2406.05839.

Guanrou Yang, Tian Tan, Qian Chen, Ziyang Ma, Yakun Song, Zhikang Niu, Qi Chen, Wenming Tu, Haitao Li, Shan Yang, and Xie Chen. GROW: Group-relative advantage-weighted on-policy reinforcement learning of autoregressive-diffusion text-to-speech model, 2026. URL https://arxiv.org/abs/2608.03215.

Yuxiang Zhao, Yichi Zhang, Yanjie An, Yanqiao Zhu, Zhanxun Liu, Yushen Chen, Qixi Zheng, Haina Zhu, Yunchong Xiao, Keqi Deng, Shuai Fan, Kai Yu, and Xie Chen. X-translator: A real-time multilingual speaker-aware speech-to-speech translation system, 2026. URL https://arxiv.org/abs/2607.17544.

# APPENDIX

This appendix expands the study counts, decision rule, and protocol details behind the main-paper diagnostics. Appendix A reconciles study units; Appendix B defines the constructed dialogue rule; Appendix C describes the benchmark partitions; Appendix D evaluates scalar ASR thresholds; and Appendices E and F give the caller details and evaluation limits.

## A STUDY POPULATIONS AND COUNTING UNITS

The replayable dialogue analyses use 3,000 source commands with one Apple and one VITS audio/ASR observation each. The retained merged record collection has 9,000 records: its two VITS exports are identical in audio path, transcript, recognizer outputs, gold call, and annotation. We remove the 3,000 repeated exports before computing these diagnostics, leaving 6,000 observations. This is a reanalysis of retained records, not new audio decoding or independent annotation validation. The two renderings of a command share its gold call and risk tier. Thus observation counts, slot counts, and command counts have different denominators.

The 925 held-out commands are a subset of this same 3,000-command cohort; the other 2,075 commands select the policy. Ten-fold selection reuses the cohort with each command evaluated once. The separate prototype, tempo, MMS-TTS, candidate-stream, and direct-audio studies are not pooled with this diagnostic. The large-v3 all-renderer support-decoder test retains 1,350 scored candidate-presentation rows from only 900 source/audio keys: 450 VITS waveforms occur with both JSON-choice and rendered-choice candidates. The gold call and risk tier agree in each pair; 159 pairs differ in candidate decision and 349 in support score. These rows measure two candidate modes on one waveform, not independent audio observations. The separate 9,000-row eSpeak-rerendered self-attention control has 9,000 distinct source/audio keys in its retained source records. Their sample sizes describe their own records or evaluation partitions; overlap with the principal cohort is unverified. Earlier caller results remain summary-level diagnostics because their predictions and original split are unavailable.

## B FORMAL DECISION PROBLEM

A spoken function-calling example contains a latent intended call $y^\star = (f^\star, a^\star)$, an acoustic observation $x$, one or more ASR observations $z$, and a dialogue action $d \in \{\text{execute}, \text{clarify}, \text{abstain}\}$. The system may output a candidate function call $\hat{y} = (\hat{f}, \hat{a})$ only when its evidence for each risk-critical argument is strong enough. Otherwise, it should ask a targeted clarification about the weakest actionable argument or abstain when no useful repair target is available.

The paper's central premise is that ASR uncertainty should be treated as uncertainty over the dialogue state, not only as uncertainty over transcript strings. Two transcripts with the same word-error rate can imply different executable arguments; likewise, a zero-edit transcript can still omit the argument support needed for safe execution after normalization, formatting, or homophone collapse. RASFC therefore evaluates policies over the action space above, not only exact-call prediction.

For a candidate call $y$, the verifier receives features summarizing posterior confidence, schema validity, candidate support, ASR-support gaps, and clean-to-ASR channel evidence. The current verifier and tier-specific controller also receive the benchmark's gold intended-action risk tier $r_i$. This tier is not inferred from the speech or selected call; the legal candidate's schema tier is a different quantity. The selective controller chooses

$$d(x, z) = \begin{cases} \text{execute}(\hat{y}) & \text{if risk threshold and support floors pass,} \\ \text{clarify}(s) & \text{if recoverable slot } s \text{ lacks clear support,} \\ \text{abstain} & \text{otherwise.} \end{cases}$$

This formulation makes the clarification target part of the prediction problem. The displayed rule describes the current oracle-risk benchmark controller, not a policy that can be deployed without first estimating the intended-action tier. Its thresholds are selected on development data and do not provide a distribution-free safety guarantee. The policy must identify which argument to ask about as well as whether to act.

Equivalently, the Bayes policy minimizes expected task loss over a belief state, not a transcript. Let $\ell(d, q, y^\star, r)$ encode the asymmetric costs used in the paper: high-risk wrong execution is much worse than low-risk wrong execution, correct targeted clarification receives partial credit, and abstention is safe but low utility. The decision rule is

risk-sensitive loss: $C_{\mathrm{hrfe}}$ is the high-risk false-execution cost, and $C_{\mathrm{hrfe}} \gg C_{\mathrm{clar}}$.

$$\pi^\star(b) = \arg\min_{d,q} \mathbb{E}_{y^\star \sim p(\cdot|x,z,\mathcal{F})}[\ell(d, q, y^\star, r)].$$

A transcript-only system replaces this posterior with a point estimate and therefore cannot distinguish three cases that require different dialogue behavior: concentrated support for a legal call, diffuse mass across several calls, and localized ambiguity in one recoverable slot. RASFC measures this distinction directly by scoring execution, targeted repair, and abstention as different actions over the same spoken command.

This can be stated as a simple decision-theoretic separation. Let $T(z)$ be any transcript-level summary used by an ASR-only policy, such as the 1-best string, WER bucket, confidence bucket, or their combination. If there exist two acoustic observations $x_1, x_2$ with the same $T(z)$ but different posteriors over risk-critical arguments, then $T$ is not sufficient for safe execution whenever the risk-aware action differs. In particular, if one posterior places mass at least $1 - \epsilon$ on the same executable call while another splits mass between two high-risk recipients, amounts, times, or destinations, then a single threshold on $T$ must either execute both or clarify both. The risk-aware policy executes the concentrated case and clarifies or abstains in the split case. Hence ASR uncertainty becomes a dialogue-state problem exactly when the missing information changes the optimal action, not merely when it changes the transcript error rate.

To make the intuition explicit, consider a simplified high-risk two-action case. Let $q$ denote the posterior probability that the selected call is correct. Correct execution has zero loss, wrong execution has loss $C_{\mathrm{hrfe}}$, and clarification has fixed loss $C_{\mathrm{clar}}$; abstention and execution reward are omitted from this special case. Then

$$\mathbb{E}[\ell \mid \textsc{execute}] = (1 - q)C_{\mathrm{hrfe}},$$
$$\mathbb{E}[\ell \mid \textsc{clarify}] = C_{\mathrm{clar}}.$$

Execution is preferred only when $q > 1 - C_{\mathrm{clar}}/C_{\mathrm{hrfe}}$. This is an illustrative threshold under the stated loss, not the empirical controller itself: the benchmark utility also includes correct-execution rewards, abstention, targeted clarification, and tier-specific costs. Two examples with identical $T(z)$ can fall on opposite sides of this threshold because $T$ discards argument-support structure. A policy using only $T$ must choose the same action for both observations, while a policy with the relevant argument evidence may choose differently. The matched-renderer audits test a related observation: equal WER/confidence buckets can have different annotated dialogue actions. They do not estimate the Bayes posterior $q$. The ASR-threshold audit reports zero correct executions for its oracle-WER thresholds at zero false execution and two true executions for its utterance-log-probability threshold at zero high-risk false execution. These operating points use a different unit and error definition from the source-group diagnostic below; their execution counts must not be compared directly with its 766 groups.

The equivalence-class diagnostic in Table 2 makes this failure mode concrete. Entries that are indistinguishable to ASR-quality buckets can still require different dialogue actions; conditioning on the supported argument set removes execute-versus-dialogue ambiguity in the current benchmark while leaving finer clarification-target ambiguity for multi-slot failures.

Table 5 aggregates this phenomenon at the source-command level. Treating matched renderer transcripts as samples from the observation channel, 353/3,000 source commands have nonzero dialogue-action entropy; 49 of those are all below 10% WER and 23 share both the WER and confidence bucket. For this offline diagnostic, `needs_dialogue` is recomputed, not independently annotated. Let $C$ mean that a gold clarification target exists, $U$ that at least one gold argument lacks transcript support, $L$ that minimum word confidence is below $0.70$ or mean utterance log probability is below $-0.45$, and $H$ that the intended risk tier is high or critical. The implemented rule is $(C \vee U \vee L) \wedge (H \vee C \vee U)$. Each rendering's action follows that rule and its gold clarification target. Oracle support, confidence, and risk therefore overlap with the target definition; information and held-out accuracy are internal-consistency measurements. The term *posterior* in the following table names the empirical fraction of those two derived actions; it is neither a fitted speech-to-action

| Diagnostic | Value |
|---|---|
| Matched source groups | 3000 |
| Volatile dialogue-state groups | 353 (0.118) |
| High-risk volatile groups | 163 (0.163) |
| All-low-WER volatile groups | 49 |
| Same WER-bucket volatile groups | 95 |
| Same confidence-bucket volatile groups | 121 |
| Clarification-target conflicts | 40 |
| Mean action entropy | 0.118 bits |

Table 5: Dialogue-state entropy over matched ASR observations. Volatile groups require different next actions for the same underlying source command, showing that ASR uncertainty changes dialogue state, not only transcript quality.

| Policy | Exec. | Clarify | Latent exec. ↓ | High-risk latent exec. ↓ |
|---|---|---|---|---|
| Max-confidence obs. | 847 | 2003 | 248 | 118 |
| Lowest-WER obs. | 847 | 1974 | 248 | 95 |
| Plurality action | 885 | 1981 | 286 | 119 |
| Posterior unanimous | 599 | 2173 | 0 | 0 |

Table 6: Posterior dialogue policy over matched ASR observations. Latent execute exposure counts source commands that the policy executes even though at least one matched observation requires clarification or abstention.

probability nor the Bayes belief $q$ above. We use *latent execute exposure* for a source command that a policy executes even though another matched rendering's derived action calls for clarification or abstention. This is a cross-renderer disagreement, not a measured incorrect real-world execution. Table 6 tests that definition: scalar best-observation policies expose up to 248 source commands to this conflict, including 118 high-risk cases. The unanimous rule executes 599 groups without such conflicts. Table 7 shows the cost of a permissive one-of-two rule: it executes 885 groups but has 286 cross-renderer conflicts.

Table 8 uses the annotated intended-action risk tier to require unanimity only for high-risk commands. It executes 766 groups, 167 more than uniform unanimity, but accepts 167 low- or medium-risk latent conflicts (87 low and 80 medium); zero high-risk conflicts does not mean zero total conflicts. Table 9 tests threshold selection on source-disjoint groups: the selected rule executes 246 of 925 held-out groups versus 206 for uniform unanimity, with 40 accepted low- or medium-risk conflicts and none in the high-risk tier. Ten-fold source-disjoint selection reports a 167-group aggregate execution lift and zero high-risk conflicts. These are offline comparisons using benchmark annotations and matched renderings, not evidence of safety for unseen human speech or an independently inferred risk tier.

Table 10 gives the complementary candidate-stream view. Before any final reranking decision, competing ASR hypotheses can disagree about whether a risk-critical waypoint was spoken; this is the operational reason the system carries candidate-level evidence into the dialogue policy instead of collapsing to a single transcript.

| $\mathbf{p}(\text{execute})$ | Exec. | Clarify | Latent exec. ↓ | High-risk latent exec. ↓ |
|---|---|---|---|---|
| 0.500 | 885 | 1981 | 286 | 119 |
| 1.000 | 599 | 2173 | 0 | 0 |

Table 7: Posterior execution frontier over matched ASR observations. Raising the posterior execute threshold monotonically reduces latent dialogue-state exposure; the unanimous point preserves non-zero execution while eliminating exposure in this offline diagnostic.

| Policy | Exec. | Latent exec. ↓ | High-risk latent ↓ | Thresholds |
|---|---|---|---|---|
| Uniform 1/2 | 885 | 286 | 119 | 0.500/0.500/0.500 |
| Uniform 1 | 599 | 0 | 0 | 1.000/1.000/1.000 |
| Risk-conditioned | 766 | 167 | 0 | 0.500/0.500/1.000 |

Table 8: Risk-conditioned posterior thresholds over matched ASR observations. Thresholds are shown as low/medium/high risk tiers. The selected risk-conditioned policy increases execution coverage over uniform unanimity while preserving zero high-risk latent dialogue exposure.

| Split | Policy | Exec. | Latent exec. ↓ | High-risk latent ↓ | Thresholds |
|---|---|---|---|---|---|
| Train | Uniform 1 | 393 | 0 | 0 | 1.000/1.000/1.000 |
| Train | Risk-conditioned | 520 | 127 | 0 | 0.500/0.500/1.000 |
| Held-out | Uniform 1 | 206 | 0 | 0 | 1.000/1.000/1.000 |
| Held-out | Risk-conditioned | 246 | 40 | 0 | 0.500/0.500/1.000 |

Table 9: Source-disjoint holdout check for risk-conditioned posterior thresholds. The policy is selected only on training source groups, then evaluated on held-out source groups; a companion 10-fold source-disjoint CV audit selects the same thresholds in every fold and preserves zero high-risk latent exposure.

Table 10: ASR candidate sets expose dialogue-state instability before reranking. An example is waypoint-volatile when competing ASR candidates disagree about whether at least one risk-critical waypoint is present.

| Diagnostic | Value |
|---|---|
| Rows | 9000 |
| Rows with waypoints | 8354 |
| Multi-candidate rows | 1299 |
| Divergent transcript rows | 1144 |
| Waypoint-volatile examples | 300 (0.036) |
| High-risk volatile rows | 142 |
| Low-gap volatile rows | 258 |
| Alt. repairs top-candidate gap | 72 |

The diagnostic is offline candidate-stream evidence, not a human-dialogue result; it motivates preserving ASR hypotheses as dialogue belief-state evidence.

Table 11 quantifies the corresponding oracle top-$k$ support ceiling: in some entries, the top ASR transcript misses a waypoint that another candidate retains. This is not a deployable oracle, but it shows that candidate sets contain recoverable dialogue-state evidence.

Table 11: Top-$k$ ASR candidates reveal an oracle support ceiling above the top transcript.

| **Diagnostic** | **Value** |
|---|---|
| Waypoint rows | 8354 |
| Multi-candidate rows | 1195 |
| Top-1 full support | 6898 (0.826) |
| Top-k full support | 6970 (0.834) |
| Top-k ceiling lift | 72 (0.009) |
| High-risk alternative repairs | 42 |

Offline candidate-stream diagnostic; it motivates preserving ASR alternatives as dialogue-state evidence without making a human-dialogue claim.

The candidate-stream feature-family summaries are retained in the private result audit, but we do not rank them against the earlier reported noisy-channel result here. The saved records do not establish common partitions and candidate streams for that comparison. The generated feature-family table therefore cannot support a paired utility difference or a method ranking in this supplement. A common-stream rerun with the same source groups and scoring protocol is needed for that comparison.

### B.1 Earlier caller and repair results

The retained noisy-channel summary reports utility 0.437 on one 1,350-example held-out partition and no executed high-risk case among 445 high-risk examples. Its per-example predictions, fitted verifier, and original split are unavailable. This summary is therefore not replayable, cannot receive a paired source-group interval, and is not used for a ranked method comparison or a headline figure. A separate ASR-candidate rescorer diagnostic reports utility 0.374 while executing 118/463 high-risk examples, with no high-risk false execution observed among those executions. The two diagnostics use different partitions and candidate streams, so they do not establish a system ranking.

The matched-risk design changes the target: the benchmark asks not whether payments are risky in general, but whether *this* amount, recipient, time, or destination is supported. On the Apple multi-voice synthetic-speech split, an action-curve diagnostic measures offline repair potential: always executing gives 0.639 HRFE; execute/abstain thresholding reaches -0.260 utility; adding clarification raises utility to 0.036. Two high-risk amount errors still appear in the raw strong-support bucket. On the 450-example Apple calibrated test subset, a development-selected value-prior targeter raises one-turn completion from 0.636 to 0.673, while utility falls from 0.131 to 0.111 and targeted-clarification precision from 0.315 to 0.235. Any zero-execution high-risk condition is reported as not estimable, not as zero error. These completion and target-precision values use oracle-risk benchmark context and annotation-based repair labels; they do not measure human dialogue behavior. Thus choosing *which* uncertainty to ask about is itself a benchmark dialogue-policy trade-off.

## C Benchmark and Splits

RASFC uses matched-risk commands across messaging, navigation, device-control, and payment-style schemas. Each command has a clean canonical call, semantic perturbations, risk tier, and rendered speech observations. The matched design asks whether a policy can preserve utility when risk-critical slots are acoustically or semantically ambiguous; it does not reward systems for avoiding whole domains.

An earlier matched-risk policy diagnostic isolates a different, within-benchmark question: how much execution remains when a development-selected tier policy is constrained to zero observed high-risk false execution. On its 1,350-example test partition, the retained tier-policy reports give utility 0.108 and high-risk execution coverage 19/449 (4.2%) without argument-evidence features. With those features, the corresponding report gives utility 0.184 and coverage 46/449 (10.2%), again with zero observed high-risk false executions. The counts are obtained by multiplying each report's

high-risk execution rate by its 449 eligible high-risk test examples; they are not independent trials. These saved summaries do not supply the original per-example predictions and fitted policies for a fresh paired replay, so the difference is a report-level diagnostic, not a verified causal feature effect or a deployment safety guarantee. The low high-risk coverage is essential to interpreting both zero-error observations. This earlier partition is not pooled with the dialogue source-group studies above or the separate audio stress tests below.

Calibration gives a further negative control on this same report-level matched-risk family. With argument-evidence features, the raw verifier has test negative log-likelihood 0.598 and expected calibration error 0.115. Development-fitted Platt scaling changes these to 0.570 and 0.101, respectively, without changing AUROC (0.756 at the reported precision). Yet its separately development-selected zero-observed-high-risk-error tier policy has utility 0.142, below the raw-score policy's 0.184; both reports retain 46/449 high-risk executions and zero observed high-risk false executions. The utility difference therefore concerns decisions outside that high-risk execution count and does not show that improved aggregate probability calibration improves the selected policy. These are retained summary-level values, not a fresh prediction-level replay or an independently estimated effect.

The saved global utility-first feature ablation asks a different question from the tier-constrained policy above. With all argument-evidence verifier features, test utility is 0.190 and high-risk false execution among executed high-risk examples is 0.107. Removing argument evidence gives 0.169 and 0.103; removing ASR confidence gives 0.157 and 0.126; removing coverage features gives 0.112 and 0.135. Removing the oracle risk-tier feature raises utility to 0.232 while high-risk false execution rises to 0.113. Thus the utility metric alone does not identify a safer configuration, and these global-threshold rows must not be read as zero-error tier-policy results. The retained aggregate reports development-selected thresholds and test summaries but not the per-example predictions needed to replay the contrasts or quantify a paired effect. We do not infer a causal contribution for any one feature from these saved summaries.

The release separates train, development, and test splits before threshold selection. Development data selects calibration, selective-execution thresholds, high-risk support floors, and clarification-target heuristics. Test data is used once for the reported frontier. The source-held-out and renderer-held-out audits stress whether decision features survive changes in wording and TTS source. The current evidence uses synthetic, rendered, and offline controlled observations. Natural human speech, listener success, clarification naturalness, human risk labels, waveform transfer, and deployment under shift are not evaluated in this study.

### C.1 Naturalistic 5k prototype and verifier-feature checks

An earlier 5,000-example naturalistic rendered-speech prototype tested Whisper `tiny.en`, `small.en`, `medium.en`, and `large-v3`. The retained project report records mean WER of 48.6%, 28.7%, 20.9%, and 16.7% in that order, while its direct transcript-to-call baseline reports high-risk false-execution rates of 44.4%, 51.6%, 55.9%, and 47.0% among executed high-risk cases. These report-level values illustrate why lower transcription error alone is not a sufficient safety measure. We do not interpret their differences as a paired causal effect of recognizer size: the original per-example predictions, split identity, and source-group joins for this prototype have not been recovered for the current paper audit.

The same 5,000-example prototype report records a `large-v3` verifier-feature ablation. Its all-feature policy has test utility 0.364 and lists high-risk false execution as 0.000, but does not retain the number of high-risk calls executed at that point. This value may therefore be a zero-coverage convention, not an estimated error rate. Removing the oracle intended-action risk-tier feature gives reported utility 0.083 and high-risk false execution 0.514. The contrast warns that the policy may depend strongly on oracle tier information; it does not establish a paired feature effect or a deployable risk estimator. The saved ablation work directory and exact split inputs named by the report are not present in this checkout, so neither the denominator nor prediction-level pairing can be recovered here. The active 9k benchmark has a different population and must not be joined numerically to this 5k prototype.

### C.2 TEMPO-AUGMENTED RECOGNITION AND ACTION SELECTION

We also test 10,000 tempo-augmented rendered examples with Whisper `medium.en` and `large-v3`. This family asks whether lower transcription error alone removes the risk of executing the wrong call. The complete-set recognizer summaries report mean WER of 22.1% and 18.2%, respectively. The direct transcript-to-call baselines still have high-risk false-execution rates of 53.3% and 48.2% among their executed high-risk cases. These are full-set baseline summaries, not results on the policy test partitions.

For the development-selected verifier policies, the saved test summaries each contain 1,500 examples: 313 high-risk examples for `medium.en` and 297 for `large-v3`. Their test utilities are 0.336 and 0.440, but both policies execute zero high-risk examples in these partitions. Consequently, the reported high-risk false-execution value of zero is a zero-coverage operating point, not an estimated error rate among executed high-risk calls. The saved summaries do not by themselves establish example-level pairing or a confidence interval for the difference in utility. This family supports the limited finding that better recognition on tempo-augmented rendered speech coexists with substantial high-risk baseline error; it does not establish high-risk selective-execution safety or transfer to human speech.

### C.3 NEURAL-SPEECH RECOGNITION STRESS CHECK

A separate balanced set of 9,000 English MMS-TTS renderings was transcribed with Whisper `medium`. The retained audit reports 3,000 examples in each risk tier, mean WER 43.7% with a bootstrap 95% interval of [43.1%, 44.3%], and 3,110/9,000 examples at WER at least 50%. Tier-specific mean WER is 36.8% for low risk, 38.7% for medium risk, and 55.5% for high risk. These are aggregate recognition results; no function-call predictions, targeted clarifications, or selective-execution actions were scored for this family. The audit cites a rendering index that is not available in the current checkout, although the saved aggregate and a second audit report agree on the principal counts. Thus this check broadens the rendered-speech recognition stress axis but does not test the paper's action policy or transfer to recordings from people.

## D ASR-THRESHOLD FRONTIER DIAGNOSTIC

The main paper argues that ASR uncertainty must be represented as dialogue-state uncertainty instead of a scalar transcript-quality score. Table 12 gives the corresponding negative control. It sweeps oracle WER, word confidence, utterance log probability, and a joint ASR-quality grid as execute gates, with outcomes defined against the constructed dialogue rule. Group-level cross-renderer exposure is reported separately because its counting unit and outcome differ. This threshold sweep characterizes a synthetic label rule; it does not establish that a dialogue policy improves independently measured execution outcomes.

Table 12: ASR-threshold frontier diagnostic. Each gate counts observations against the constructed dialogue rule. True execution means execution where that rule does not require dialogue; HR false counts high-risk observations where it does. These outcomes differ from incorrect-call HRFE. Source-group exposure is reported separately in Table 8.

| Policy | Constraint | Unit | Exec. | True exec. ↑ | HR false ↓ |
|---|---|---|---|---|---|
| Oracle WER | zero false exec. | example | 0 | 0 | 0 |
| Utterance log probability | zero high-risk false exec. | example | 11 | 2 | 0 |
| Joint ASR grid | zero high-risk false exec. | example | 0 | 0 | 0 |

Best ASR-only gates recover 0 true executions at zero false execution and 2 at zero high-risk false execution.

## E NOISY-CHANNEL CONSTRAINED CALLER

The noisy-channel caller scores legal function-call candidates instead of free-form text. It expands schema values into spoken aliases, estimates how likely each clean argument could produce the observed ASR tokens, and combines these scores with candidate-level confidence and verifier fea-

tures. This yields argument-level evidence that can be audited by slot, risk tier, perturbation type, and renderer.

1. Enumerate legal candidates from the function schema and domain inventory.
2. Normalize clean values and ASR observations into comparable spoken forms.
3. Estimate token-channel support for each candidate argument.
4. Select a candidate call using schema validity plus aggregate channel evidence.
5. Calibrate a verifier on development data and apply development-selected thresholds to test data.
6. Route low-risk unsupported cases to targeted clarification and high-risk unsupported cases to clarification or abstention.

This constrained scoring differs from transcript correction. It asks whether the observed speech supports a particular executable argument, which is the evidence needed for dialogue policy.

Table 13: Information access for the caller families. Candidate surfaces are legal schema candidates or their supplied speech renderings. Clean reference transcripts in the training column are never test-time inputs.

| **Caller family** | **Training inputs and supervision** | **Test-time inputs** |
|---|---|---|
| Transcript-only constrained | Train transcripts and schema; gold call labels and train/dev verifier-action labels | 1-best ASR transcript and schema inventory |
| ASR-candidate rescorer | Candidate streams and diagnostics; call labels and train/dev candidate-action labels | ASR $n$-best observations, confidence features, and schema inventory |
| Noisy-channel constrained | Paired clean/source and ASR transcripts; gold calls for channel/caller fitting and train/dev verifier-action labels | ASR observation, schema inventory, and learned channel parameters |
| Prompt-only direct audio | Fixed model and prompts; in-context instructions or examples, without task-specific parameter updates | Waveform and prompt with legal candidate choices |
| Waveform/candidate-inventory support decoder | Waveform/audio-LALM-derived candidate inventory; exact-match candidate-support labels and dev-selected policy thresholds | Waveform-derived embedding, legal candidate metadata, and benchmark perturbation indicators in current code; report-to-code match unverified |

| **Caller family** | **Candidate surfaces** |
|---|---|
| Transcript-only constrained | Legal templates and argument values |
| ASR-candidate rescorer | Legal candidates and observed ASR candidate surfaces |
| Noisy-channel constrained | Legal candidates with spoken aliases |
| Prompt-only direct audio | Candidate wording and/or JSON choices supplied in the prompt |
| Waveform/candidate-inventory support decoder | Legal candidates supplied by the candidate inventory |

Table 13 makes the information boundary explicit across the caller families. A paired clean reference transcript is used only as train-split supervision for the noisy-channel parameters; it is unavailable to every caller and policy as a test-time observation. The current direct-audio support-decoder code consumes waveform-derived embeddings, legal candidate metadata, and annotation-derived perturbation indicators at test time. Those saved result summaries have not been matched to that code, so their input boundary remains unverified.

## F Metrics and Evaluation Scope

The primary safety metric is high-risk false execution: executing an incorrect high-risk call. Accuracy and slot F1 remain useful diagnostics, but they are insufficient because they do not distinguish a harmless mismatch from an unsafe payment, message, route, or device action. Utility rewards correct execution, assigns lower value to clarification, and penalizes unsafe execution.

The reported safety frontier is limited to synthetic, rendered, and offline observations. Natural human speech, listener success, clarification naturalness, human risk and clarification labels, waveform transfer, and deployment under shift are outside the present evaluation. The candidate-stream and rescoring controls test whether richer ASR evidence can replace argument-level support; they do not establish human-dialogue performance.