AdaSpeechRAG: Auditing Evidence for Supplied-Context ASR

Anonymous authors
Paper under double-blind review

Abstract

Automatic speech recognition (ASR) systems must sometimes decide whether to use contextual hints or a more expensive recognizer. Applying the strongest recognizer to every utterance wastes computation, and indiscriminate context can insert unsupported words. Prior work usually evaluates these actions separately, which makes their quality, latency, and risk trade-offs difficult to compare. We introduce AdaSpeechRAG, an auditable interface for a supplied-context decision: keep a cheap transcript or add a supplied entity prompt. On the 3,750-utterance held-out partition of ContextASR-Bench, fixed prompts reduce word error rate (WER) from 26.66% to 22.11% and improve entity-level F1 from 66.15% to 83.98%. A rule chosen on the training and development partitions reaches 21.94% WER while prompting 60.27% of test utterances. Relative to fixed prompting, its paired absolute character error rate (CER) difference is −0.37%; prompt use and recorded p95 real-time factor (RTF) also fall in one execution setting. Its WER difference remains unresolved, while entity-level F1 falls to 82.96%. These matched comparisons expose a quality, entity, and prompt-use trade-off that a WER-only summary would miss. AdaSpeechRAG records the scope and provenance of each action so that supplied-context results and stored action counts remain distinguishable.

1 Introduction

Automatic speech recognition (ASR) systems encounter a small but consequential set of difficult inputs, including dialectal speech, rare entities, overlapping speakers, long utterances, and code-switching. These inputs create a context decision: expose hints that may help with rare words but can also insert unsupported text, or retain the cheap transcript. The practical problem is to decide when supplied context is useful and to measure its benefit against recorded latency and provenance.

Selective-invocation work studies when to run a stronger recognizer (Mu et al., 2025a; Xue et al., 2025), while contextual ASR studies how vocabulary or side information changes recognition (Lin et al., 2025; Mu et al., 2025b; Tian et al., 2026b). Expert routing, temporal grounding, alignment, and speaker-attributed recognition add related mechanisms and safety concerns (Wang et al., 2025a; Mu et al., 2025c; 2026a; Gao et al., 2026a; Dai et al., 2026; Lin et al., 2026; Mu et al., 2026b; Gao et al., 2026b). The closest selective-invocation family varies an escalation action; the closest contextual-ASR family varies an input-side context action. AdaSpeechRAG does not claim a new decoder, a new gate architecture, or a shared policy over those families. Its distinct object is an action-specific admission record that binds each reported action to its input and provenance state, split rule, metric and uncertainty, latency environment, and permitted finding. This record does not make scores across datasets commensurate or act as a same-audio controller; it makes the missing conditions for such a comparison visible. The question is simple: can an admission record prevent a supplied-context evaluation from being generalized beyond the evidence it contains?

Table 1 states the resulting division of labor. Selective invocation asks whether escalation helps on a task; contextual ASR asks whether a specified context representation helps recognition. AdaSpeechRAG asks a separate question: whether the evidence packet admits the

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.

Table 1: Conceptual positioning against the two closest research families. This table compares the question and output of each family; it does not compare methods or claim that any prior method lacks an audit procedure.

Research family	Varied action	Question	Output in this paper
Selective invocation	Escalate from a cheaper to a stronger recognizer	Does escalation improve the stated task under its decision and budget protocol?	An archived Wenet/WSC action-accounting case, without a regenerated performance conclusion.
Contextual ASR	Supply a vocabulary, sidecar, or other specified context	Does the specified context improve recognition on the stated evaluation unit?	A ContextASR supplied-entity frontier and an FSC negative guardrail.
AdaSpeechRAG	Record either action as a separate packet	Does the packet contain the fields needed for the requested finding?	An admission receipt that permits only the conclusion supported by provenance, split, metric, uncertainty, and resource records.

stated scope of either result. The comparison is conceptual, not a performance baseline or a claim that prior methods lack audit procedures.

We answer this question with AdaSpeechRAG[1], an audit interface that records the input version, cheap transcript and features, selected action, context source and provenance state, decoder/checkpoint, latency environment, recognition outcome, and allowed finding. In the evaluated ContextASR setting, cheap denotes the small faster-whisper configuration and strong denotes large-v3-turbo. The prospective experiment instantiates two actions: keep the cheap transcript or add a benchmark-supplied entity list. A separate external archive records earlier stronger-recognizer actions, but its missing prediction shards block a new decoder-performance finding. The interface also reserves repair and abstention for weak or unsafe evidence, but the paper does not score those outcomes as learned gains. This separation keeps the evaluated controls visible without presenting an unevaluated controller.

The interface is an audit contract, not a common estimand or globally identified utility. Each action family retains its own population, independent unit, split rule, metric, and decision rule. AdaSpeechRAG aligns the fields required to inspect an action, but it does not rank results across datasets or assert that the reported policies form one jointly trained controller.

The evaluation connects each action to a concrete question. ContextASR-Bench tests whether a selector chosen on training and development data retains the benefit of supplied entity prompts on held-out speech while using fewer prompts. Wenet-Wu and WSC-Eval supply an archived action-accounting case whose stored trace is checked against its aggregate counts, but does not test decoder performance anew. Fluent Speech Commands (FSC) provides a negative guardrail: ontology prompts lower raw WER but worsen normalized error. Together, these settings test the interface while preserving the distinct meaning of each input and split.

Contributions. Our contributions are threefold.

- We define an auditable record for cheap-pass acceptance and supplied-context prompting, with explicit latency, recorded resource, and provenance terms. It is an audit schema, not a cross-action optimizer. Repair and abstention remain identifiable contract outcomes instead of being reported as measured improvements.

[1]Code for reproducing the reported evaluations is provided as supplementary material under the MIT License.

- We report prospective ContextASR-Bench selection under a fixed held-out protocol. Its results expose a quality–prompt–latency trade-off instead of reducing performance to one accuracy number.
- We implement a fixed admission policy that checks packet identifiers, required fields, and locked artifacts. It permits the supplied-sidecar frontier, retains an external archive for recorded action accounting only, and blocks regeneration and natural-context transfer conclusions when required records are absent.

2 AdaSpeechRAG Decision Interface

Let x_i be an utterance, $\hat{y}_{c,i}$ the cheap transcript, r_i a set of retrieved or sidecar context candidates, and z_i runtime features from the cheap pass. In this study, z_i starts with cheap confidence summaries, low-cost acoustic evidence, and difficulty, then adds alignment quality plus speaker-turn, overlap, and channel cues whenever the underlying metadata can support them. AdaSpeechRAG records an action

$$a_i \in \underbrace{\{\text{accept}, \text{context}\}}_{\text{evaluated controls}} \cup \underbrace{\{\text{strong}, \text{repair}, \text{abstain}\}}_{\text{archival or specified outcomes}} \tag{1}$$

whose quality, cost, latency, and provenance terms are recorded as:

$$\begin{aligned} \mathcal{L}(a_i) = \widehat{e}(a_i, z_i, r_i) + \lambda_c C(a_i) + \lambda_l T(a_i) \\ + \lambda_h H(a_i, r_i) + \lambda_p P(a_i, r_i). \end{aligned} \tag{2}$$

Here C is an optional recorded resource proxy, T is latency measured as real-time factor (RTF), H is prompt-harm risk, and P is unsupported-context risk for actions that expose retrieved evidence. The reported prospective study records prompt use and measured shard RTF; it does not report one portable compute measure across settings. The other recorded fields are cheap-transcript phrase coverage, duration, confidence statistics, source labels, and bootstrap-estimated recognition deltas. The equation is a bookkeeping interface, not an end-to-end objective optimized by a single multi-action model in this paper. A stronger-recognizer action remains a schema slot whose performance requires a matched packet.

Admission rule. For a packet $P = (F, D)$ with available fields F and locked artifact digests D, a claim q is permitted exactly when its predeclared required-field set R_q satisfies $R_q \subseteq F$ and every digest named by q matches D. Otherwise the receipt returns blocked with $R_q \setminus F$. For example, the supplied-sidecar frontier requires benchmark sidecar, matched held-out audio, train/development selector, paired uncertainty, p95 RTF, and utterance split; all are present. Decoder regeneration on the Wenet/WSC archive requires paired prediction shards, which are absent and therefore blocked. The validator fixes packet identifiers, claim identifiers, field sets, and artifact hashes independently of the receipt, so a receipt cannot widen its own scope.

The prospective decoders instantiate the accept and context actions. The strong action appears only in an archival record. Repair and abstention are explicit safety outcomes in the policy contract, but this paper does not present them as learned performance gains because the required natural-context, overlap, and timestamp evidence is unavailable in the evaluated packets.

Executable selectors. The ContextASR fixed p70 reference applies the frozen p70 (70th-percentile) difficulty threshold before exposing the top-k sidecar entity phrases. The train/dev pair rule searches deployable binary gates over empirical duration quantiles $q \in \{0.60, 0.65, 0.70, 0.75, 0.80, 0.85, 0.90\}$ and feature quantiles $q \in \{0.25, 0.35, 0.45, 0.50, 0.60, 0.70, 0.80\}$. Its selected test-time gate accepts the contextual action when either duration is below the selected duration threshold or prompt character density exceeds the selected threshold. This duration direction is empirical: development slices show a prompt-induced CER regression on the longest utterances, so the gate suppresses context above the selected cutoff. The evaluated configuration fixes those thresholds at 54.33 seconds and 2.9971811150506786 characters/second.

Table 2: Action-specific evidence and the conclusion it permits. The shared record makes the decision interface comparable; it does not turn distinct data, split, or provenance conditions into one jointly evaluated controller.

Action setting	**Verified record**	**Permitted conclusion**
Supplied entity prompting (ContextASR-Bench)	Benchmark entity sidecar, train/development policy selection, matched held-out audio, paired bootstrap, and p95 RTF.	A supplied-sidecar quality–prompt-use–latency frontier on this held-out partition.
Archived actions (Wenet-Wu, WSC-Eval)	Aggregate fold-level summaries, feature schema, fold-specific thresholds, metric counts, and a stored per-decision action trace with recorded chunk seconds; original paired prediction shards are unavailable.	An archival action-accounting case; not evidence that a routing policy improves ASR, and not regeneration, natural-context, or session-disjoint transfer evidence.
Ontology/history controls (FSC, ASCEND)	FSC group-disjoint point-estimate guardrail; ASCEND same-session diagnostic with incomplete alignment and speaker-risk record.	A negative guardrail or inconclusive diagnostic; neither supports natural-context, repair, abstention, or speaker-aware claims.

We also fit a higher-capacity control to test whether the simple pair rule leaves an obvious modeling gain. Independent ridge regressors predict the utility of the no-context and fixed-p70 candidates from cheap-pass confidence, duration, transcript-length, and prompt-count features. The per-utterance utility combines normalized word and character error with entity recall; no latency or action-cost penalty is used. This is a recognition-focused capacity control, not a fully cost-tuned joint policy. Training uses 7,500 examples, the margin threshold is chosen on 3,750 development examples under a 75% cap on fixed-p70 candidate selection, and recognition outcomes are computed once on the remaining 3,750 examples.

The development objective lexicographically minimizes CER and WER, maximizes entity F1, and then minimizes prompt harm and context rate. The joint-cost variant uses the same candidate family with recorded weights $1\times$WER $+1\times$CER $-0.25\times$entity F1 $+0.05\times$harm $+0.01\times$context $+4\times$p95 RTF. The external archive records a fold-specific linear-score procedure over cheap-pass features, but its paired prediction shards are unavailable. It is therefore used only to verify stored action accounting. These interfaces are separate, not one jointly trained policy.

Figure 1 separates three evidence states: a matched supplied-sidecar evaluation, an archived action trace without paired predictions, and an FSC ontology control whose raw and normalized error metrics move in opposite directions. These are separate studies, not branches of one evaluated controller. Table 2 makes the admission rule explicit: each action family carries the record needed for its scoped finding, and missing fields block broader conclusions. The ContextASR pair rule records a benchmark-supplied entity sidecar, train/development selection, held-out recognition, and p95 RTF, so it permits a supplied-sidecar frontier finding. The Wenet/WSC archive records 9,130 held-out cases, fold assignments, thresholds, metric counts, and a stored per-decision action trace. The original paired prediction shards are unavailable, and the archive lacks natural-context inputs and source-session manifests. The trace therefore supports inspection of recorded actions and their aggregate counts, but not end-to-end regeneration, natural-context, or session-transfer findings.

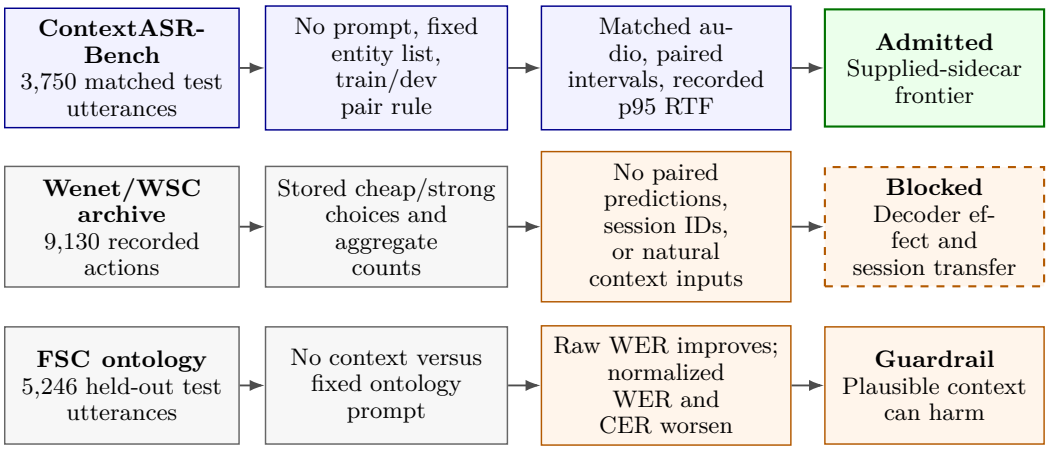

Figure 1: Three evidence states in AdaSpeechRAG. The matched ContextASR-Bench comparison supports a supplied-sidecar quality and prompt-use frontier. The Wenet/WSC archive supports stored action accounting, but unavailable paired predictions and source-session records block decoder-performance and transfer conclusions. On the later FSC test split, fixed ontology context lowers raw WER while increasing normalized WER and CER, so it is a negative guardrail. Arrows connect only fields from the same study; they do not represent one jointly trained controller.

Admission audit in practice. The record changes what each result can support. A machine-readable receipt is checked against a fixed policy with three named packet types, fixed required fields, and locked-artifact digests; it rejects a receipt that changes its own required fields or packet identity. The retained Wenet/WSC trace checks the action counts in the appendix's external action-accounting table, but missing paired prediction shards bar regeneration and missing source-session manifests bar session-transfer claims. The ASCEND packet lacks alignment and speaker-risk fields, so it is retained only as an inconclusive diagnostic. On FSC, raw WER improves while normalized WER/CER worsen, so the contextual condition is a guardrail instead of a positive transfer result. These are testable scope decisions, not post-hoc caveats.

Cheap-pass signals. The first pass uses a compact recognizer to obtain transcript text, timestamps when available, confidence-like decoding statistics, duration, no-speech indicators, and text-unit counts. For contextual experiments, phrase/entity features measure how much of the sidecar evidence is already covered by the cheap transcript. The archival record stores comparable cheap-pass fields but cannot support a new stronger-recognizer result.

Context activation. Context is treated as an action, not a default. The policy may pass no context, fixed sidecar prompts, source-filtered prompts, prompt-density rules, or learned utility gates to the strong decoder. When provenance, overlap, timestamp, or split-leakage checks fail, the policy resolves the case as repair or abstention, not as another context action. This matters because context can improve rare entities while also over-biasing acoustically plausible text when evidence is weak.

Stronger-recognizer and repair records. The external archive records faster-whisper small and large-v3-turbo action fields. Its matched prediction-level inputs and older multi-tier materials are unavailable, so the paper uses it only as an example of a blocked stronger-recognizer finding. A future packet could support stronger-recognizer selection only after it supplies matched audio, predictions, resource measurements, and the fields required for the intended conclusion.

3 Experimental Protocol

We use several settings with different evidentiary roles. Multilingual FLEURS provides a controlled difficulty-calibration test across scripts and languages (Conneau et al., 2023); it is reported only as an appendix diagnostic and does not support a headline finding. ContextASR-Bench supplies entity-heavy English speech with benchmark-provided entity sidecar phrases, enabling WER, CER, entity recall/F1, and prompt-harm analysis (Wang et al., 2025b). The sidecar is copied from the benchmark entity list, whose phrases occur in the reference transcript; this setting is a supplied-entity control, not natural retrieved session context. Its fixed policy packet contains 7,500/3,750/3,750 train/development/test utterances with disjoint unique utterance keys, but no speaker, session, recording, or document identifier. The packet therefore cannot support group-held-out or within-group feature analyses; its 21 source prefixes occur in every split and are not treated as a group split. WenetSpeech-Wu-Bench and WSC-Eval provide archived dialect-ASR action records (Wang et al., 2026a; Dai et al., 2025). Their source-session manifests, natural-context inputs, and paired prediction shards are unavailable. The retained trace therefore supports recorded action accounting only, not new decoder evaluation, multi-tier routing, retrieval-augmented recognition, or session-disjoint transfer. We also retain a separate ASCEND natural-session-history decode as a bounded diagnostic. It uses non-oracle same-session history, but its packet fails the paper-facing checks for short-term-memory/turn alignment, alignment quality, overlap/channel, and speaker-risk metadata. The evaluated configurations and result-table mappings are fixed in the accompanying reproducibility release.

The ContextASR policy follows a prospective train/development/test protocol. The external archive uses fixed utterance-key cross-fit and stores one learned-gate action record per held-out utterance, while the original paired prediction shards needed for regeneration are unavailable. Its paired intervals are preserved as recorded aggregate metadata, not recomputed decoder evidence. Audits check input integrity, leakage controls, split disjointness where source groups exist, LaTeX builds, and run completeness before results are reported. Real-time factor (RTF) is measured wall-clock decoding time divided by audio duration under each recorded execution setting; it is a within-setting latency proxy, not a portable hardware-independent compute measure. The protocol asks two evaluated questions: (Q1) does supplied sidecar context improve the same held-out audio; and (Q2) can a train/dev gate retain that gain with fewer prompts? It also checks whether the external archive's stored actions agree with its aggregate record. Natural-context repair, abstention, joint context-plus-expert selection, and decoder-routing efficacy are outside these tests.

4 Results

4.1 Supplied Entity Prompts Help, but Selection Changes the Trade-off

On the 3,750-utterance ContextASR-Bench held-out partition, fixed p70 prompting establishes the supplied-context baseline. Held-out WER falls from 26.66% to 22.11%, CER falls from 9.05% to 8.22%, and entity F1 rises from 66.15 to 83.98. The appendix budget sweep shows the expected cost: WER improves more as prompt use rises.

Table 3 compares the held-out rules. The training/development pair rule yields 21.94% WER, 7.85% CER, and 60.27% prompt use. Against fixed p70, its paired absolute CER difference is -0.37% (95% CI $[-0.71\%, -0.03\%]$); the absolute WER difference is -0.17% (95% CI $[-0.57\%, +0.23\%]$), so a WER improvement is unresolved. The absolute entity-F1 difference (pair rule minus fixed p70) is -1.01% (95% CI $[-1.19\%, -0.84\%]$). Prompt use falls from 64.56% to 60.27%, and recorded p95 RTF falls from 0.0444 to 0.0361 in one execution setting. Figure 2 shows this quality–entity–latency trade-off. A joint-cost rule uses 52.88% prompts and approaches the no-context RTF in that setting, with higher WER and lower entity F1 than the pair rule. The appendix gives paired intervals against no context. No displayed rule is uniformly best.

Table 3: Main ContextASR-Bench results. Policies are selected without test labels and evaluated on matched held-out predictions. The fixed and pair-rule entries come from prospective runs; the learned utility selector replays the same no-context and fixed-p70 candidates after training on the training split and threshold selection on the development split. Prompt use is the percentage of utterances with a nonempty context string. p95 RTF is a within-setting wall-clock latency proxy, not portable compute. The companion appendix reports paired 95% bootstrap intervals from 2000 utterance resamples. Lower WER, CER, prompt use, and p95 RTF are better; higher entity F1 is better.

Policy	WER (%) ↓	CER (%) ↓	Entity F1 (%) ↑	Prompt use (%) ↓	p95 RTF ↓
No context	26.66	9.05	66.15	0.00	0.0255
Fixed p70 context	22.11	8.22	83.98	64.56	0.0444
Learned utility selector	22.44	8.31	83.16	60.64	0.0377
Train/dev pair rule	21.94	7.85	82.96	60.27	0.0361
Joint-cost pair rule	22.76	8.14	81.01	52.88	0.0263

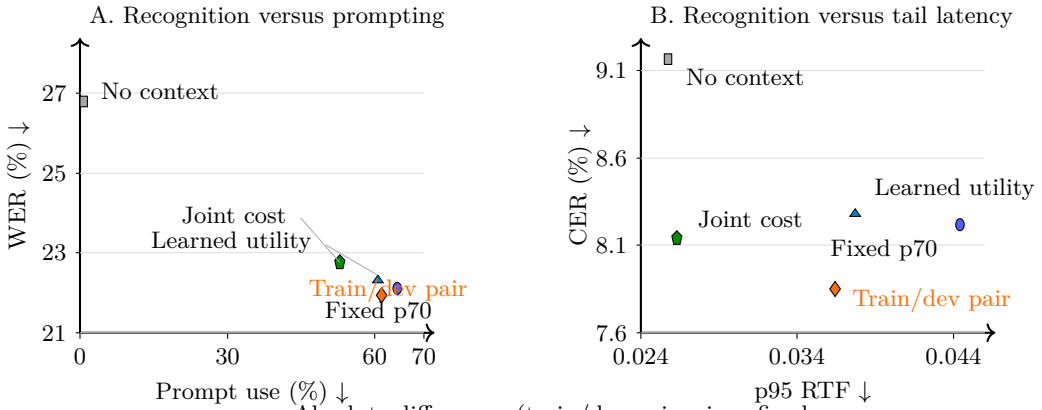

Figure 2: Measured ContextASR-Bench quality-cost frontier on the shared 3,750-utterance held-out partition. Panel A relates prompt use to WER; Panel B relates p95 real-time factor (RTF) to CER. Lower values are better on every axis. The train/dev pair rule lowers CER, prompt use, and p95 RTF versus fixed p70, but its WER interval includes zero and its absolute entity-F1 difference is −1.01% (pair rule minus fixed p70). The points are measured table values, not fitted trade-off curves.

The learned utility control does not improve this frontier. It prompts 60.64% of test utterances, reaches 22.44% WER and 8.31% CER, and retains an entity F1 score of 83.16. For learned utility minus fixed p70, its absolute prompt-use difference is −3.92%, while p95 RTF falls from 0.0444 to 0.0377, WER rises by 0.33% (paired 95% interval [0.20%, 0.49%]), and entity F1 falls by 0.82%. It also has higher WER, CER, prompt use, and tail latency than the pair rule, with an absolute entity-F1 difference of +0.20% (learned utility minus pair rule). This negative control does not compare a fully cost-tuned learned policy.

Table 4: Public FSC ontology-context aggregate point estimates on the later group-disjoint held-out test split ($N = 5{,}246$). Context phrases are a global train-split task ontology, not transcript-derived oracle text. The exact paired predictions are unavailable for metric replay or uncertainty estimation. Lower WER/CER and p95 RTF are better.

Policy	Ctx. (%) ↓	Raw WER (%) ↓	Norm WER (%) ↓	Norm CER (%) ↓	p95 RTF ↓
No context	0.00	26.48	4.54	2.21	0.1084
Fixed ontology context	39.52	21.69	4.76	2.43	0.1072

The ContextASR evidence also exposes failure modes. Duration caps can reduce tail latency but may hurt long-form CER. Prompt-harm oracles remain much stronger than learned harm gates. Unsupported context can harm WER/CER even when it looks semantically related. These diagnostics justify retaining provenance and harm penalties in the policy objective; retrieval is not uniformly beneficial.

4.2 An External Record Can Be Auditable Without Being a Result

The appendix preserves a Wenet/WSC action-accounting record under utterance-key modulo cross-fit. Its trace and aggregate agree on stored action counts, but paired prediction shards, source-session manifests, and natural-context inputs are unavailable. The archived CER fields therefore do not establish decoder performance or transfer; they demonstrate why the admission policy labels the corresponding stronger-recognizer finding as blocked.

4.3 Ontology Context Is a Negative Guardrail

As a benchmark-specific guardrail, Table 4 stages Fluent Speech Commands as a group-disjoint ontology benchmark. On the later held-out test split ($N = 5{,}246$), fixed ontology context lowers raw WER from 26.48% to 21.69%, but worsens normalized WER/CER from 4.54%/2.21% to 4.76%/2.43%. The saved aggregate supports these point estimates, but the exact paired prediction files are unavailable for replay or an uncertainty interval. The completed dev/train/test summaries are inconsistent across splits: development slightly prefers the ontology prompt, but train and held-out test prefer no context. The development-selected condition matches fixed ontology context on the held-out test split, so it does not demonstrate a separate selection gain. The earlier FSC metric-conflict analysis in the appendix uses a distinct 4,371-example test split. Without the two split files, their example overlap is unknown and their utterance counts and paired differences cannot be combined. We retain FSC ontology context as an aggregate-backed, benchmark-specific guardrail, not evidence for natural conversational or egocentric contextual-RAG transfer.

A separate ASCEND natural-session-history packet provides an inconclusive diagnostic, not headline contextual-transfer evidence. The packet contains 12,310 same-session utterance records with no transcript-derived context leakage. Its history and matched no-context decodes use the same 80.84% strong-model escalation rate. History changes raw WER/CER from 75.45%/27.88% to 74.58%/27.65%; the paired absolute differences (history minus no context) are -0.86% in WER (95% CI $[-2.13\%, 0.51\%]$) and -0.24% in CER ($[-1.09\%, 0.68\%]$). Under the current phrase-evidence matching protocol, neither condition recovers a reference-matched entity, so entity recall is zero in both. The packet lacks the alignment and speaker-risk fields required by the target action contract, and the intervals do not establish a reliable quality gain. It therefore does not close the natural-context evaluation gate or support repair, abstention, or speaker-aware statements. Its evaluation details are retained with the reproducibility record.

The admission policy yields a deployment caveat: source-side cheap-pass scores and thresholds cannot substitute for a target packet with matched actions and resource measurements. In practice, collect cheap-pass scores first, define the target-side evidence packet, and only then evaluate expensive decoding.

5 Discussion

The results answer a narrower and more useful question than whether one controller solves every ASR condition. Supplied entity prompts and stronger recognizers require different records. ContextASR-Bench exposes a quality–entity–latency trade-off, the Wenet/WSC archive demonstrates action accounting without a new decoder claim, and the FSC guardrail shows that even a plausible ontology prompt can worsen normalized error. The shared action record keeps these outcomes distinct instead of treating them as interchangeable demonstrations of natural context.

The utility-regression control sharpens this interpretation. A more flexible selector recovers most of the fixed-context gain, yet it fails to surpass the simpler pair rule on recognition quality, prompt use, or tail latency. Selector capacity alone is therefore not the missing ingredient in this recognition-focused setting; trustworthy features, matched action semantics, and transparent validation remain central.

Limitations. The Wenet/WSC archive is an action-accounting case, not a regenerated dialect-routing result, and the FSC condition is an ontology-context negative control; neither evaluates natural contextual retrieval. The ASCEND history diagnostic also lacks the alignment, overlap, channel, and speaker-risk records required by the action contract, and its uncertainty intervals include no change. In addition, the external archive uses utterance-key splits because source-session manifests are unavailable. These facts restrict the findings to the named packets and protocols.

The next natural-context study has a precise admission rule. It must compare cheap no-context, matched fixed context, stronger-recognizer invocation, all-strong with fixed context, and multi-action selection on the same audio. It must also record timestamp alignment, speaker evidence, context provenance, action rates, latency, and failure slices. The current data package lacks these records as well as deployment labels for privacy, consent, and speaker-level disparity, so no natural-context or high-stakes deployment result is reported here.

Responsible deployment requires the same conservatism. AdaSpeechRAG should not be used as an unsupervised high-stakes transcription system: retrieved phrases can amplify private or stale context, speaker-attribution errors can misassign utterances, and selective strong decoding may unevenly allocate accuracy across speakers or dialects. These allocation effects are not measured here. We therefore treat provenance labels, leakage audits, calibration curves, and per-slice error reports as release gates.

6 Conclusion

AdaSpeechRAG contributes an auditable way to evaluate supplied-context ASR. On held-out ContextASR-Bench speech, a training/development-selected rule lowers CER and prompt use relative to fixed p70, while its WER difference is unresolved and its entity F1 is lower; its p95 RTF difference is reported for one evaluated environment. The Wenet/WSC archive passes only an action-accounting check: it retains stored action counts and aggregate fields, while missing prediction shards block a routing-performance conclusion. The negative FSC result is equally important: plausible context does not guarantee better normalized recognition. Together, these findings show why supplied context needs a stated selection rule, matched comparisons, and provenance records.

For this recognition-focused objective, increasing selector flexibility does not by itself improve the measured trade-off. The paper's contribution is therefore the auditable decision interface and the evidence needed to accept or reject each action policy, not a promise that a more complex gate will always help.

The conclusion is limited to supplied entity prompts. A joint natural-context controller or recognizer-routing study needs same-input actions, group-aware splits, and alignment, speaker, prediction, and resource records absent here. These requirements define the next evaluation without implying that it has been completed.

AI Use

AI tools were used to help polish the writing and to retrieve and discover literature, including related work. The paper describes these uses.

Reproducibility Statement

The supplementary source archive provides evaluation code and configurations under the MIT License. Reproducing the reported metrics also requires the cited benchmark data and model checkpoints, which are not redistributed in the archive.

References

B. Mu et al., "Efficient Scaling for LLM-based ASR," in Proc. ASRU, 2025.

W. Tian et al., "dLLM-ASR: A Faster Diffusion LLM-based Framework for Speech Recognition," arXiv:2601.17902, 2026.

L. Li et al., "Audio-Cogito: Towards Deep Audio Reasoning in Large Audio Language Models," arXiv:2604.12527, 2026.

H. Xue et al., "Selective Invocation for Multilingual ASR: A Cost-effective Approach Adapting to Speech Recognition Difficulty," in Proc. Interspeech, 2025.

Z. Lin et al., "Contextualized Automatic Speech Recognition with Dynamic Vocabulary Prediction and Activation," in Proc. Interspeech, 2025.

B. Mu et al., "Hearing More with Less: Multi-Modal Retrieval-and-Selection Augmented Conversational LLM-Based ASR," arXiv:2508.01166, 2025.

W. Tian et al., "Seeing the Context: Rich Visual Context-Aware Speech Recognition," arXiv:2603.07263, 2026.

H. Wang et al., "CAMEL: Cross-Attention Enhanced Mixture-of-Experts and Language Bias for Code-Switching Speech Recognition," in Proc. ICASSP, 2025.

B. Mu et al., "HDMoLE: Mixture of LoRA Experts with Hierarchical Routing and Dynamic Thresholds for Fine-Tuning LLM-based ASR Models," in Proc. ICASSP, 2025.

B. Mu et al., "LLM-ForcedAligner: A Non-Autoregressive and Accurate LLM-Based Forced Aligner for Multilingual and Long-Form Speech," arXiv:2601.18220, 2026.

D. Gao et al., "LAT-Audio: Listening with Time: Temporal-Aware Long Audio Understanding," arXiv:2604.22245, 2026.

Y. Dai et al., "TAG-Bench: Benchmarking Temporal Audio Grounding in Large Audio Language Models," arXiv:2609.00752, 2026.

Z. Lin et al., "Speaker-Reasoner: Scaling Interaction Turns and Reasoning Patterns for Timestamped Speaker-Attributed ASR," arXiv:2604.03074, 2026.

B. Mu et al., "DiaScriber: A Speech LLM for Joint Diarization and Transcription in Multi-Speaker Scenarios," arXiv:2608.22796, 2026.

D. Gao et al., "The SLT 2026 SmartGlasses Challenge: Benchmarking Egocentric Multi-Talker Speech Recognition and Understanding with Audio-Language Models," arXiv:2608.12034, 2026.

A. Conneau et al., "FLEURS: Few-shot Learning Evaluation of Universal Representations of Speech," in Proc. SLT, 2023.

H. Wang et al., "ContextASR-Bench: A Massive Contextual Speech Recognition Benchmark," arXiv:2507.05727, 2025.

C. Wang et al., "WenetSpeech-Wu: Datasets, Benchmarks, and Models for a Unified Chinese Wu Dialect Speech Processing Ecosystem," arXiv:2601.11027, 2026.

Y. Dai et al., "WenetSpeech-Chuan: A Large-Scale Sichuanese Corpus with Rich Annotation for Dialectal Speech Processing," arXiv:2509.18004, 2025.

S. Wang et al., "On-Policy Self-Distillation for Multi-Dialect ASR: Mastering Dialects, Retaining Mandarin," arXiv:2608.11898, 2026.

Q. Liao et al., "Summary of the ChinaVoices Challenge 2026: Data, Tasks, Baseline, and Methods," arXiv:2609.03471, 2026.

AdaSpeechRAG Appendix

Anonymous authors
Paper under double-blind review

This appendix supplies protocol and supporting measurements for the compact conference paper. The central result is the held-out ContextASR-Bench supplied-sidecar frontier. The Wenet/WSC material is retained only as an archived action-accounting case: it preserves a trace and aggregate record, but not the paired prediction shards required for a regenerated decoder result. Repair, abstention, and joint context-plus-expert selection are evaluation targets, not completed empirical results.

The action framing follows work on low-latency and efficient automatic speech recognition (ASR), contextual vocabulary and retrieval, expert routing, and structured speech evaluation. Efficient-routing studies examine selective invocation and expert specialization, including Efficient Scaling (Mu et al., 2025a), HDMoLE (Mu et al., 2025c), and Selective Invocation (Xue et al., 2025). Related systems study delayed decoding and contextual ASR (Li et al., 2025; Lin et al., 2025), as well as hearing-related context, forced alignment, and structured speech evaluation (Mu et al., 2025b; 2026a; Tian et al., 2026a; Wang et al., 2025a; Lin et al., 2026).

1 FLEURS Calibration Diagnostics

The FLEURS experiments (Conneau et al., 2023) show why the method does not rely on a single global difficulty threshold. Tables 1, 2, 3, and 4 document early calibration failures. High thresholds produced zero escalation; script-aware recalibration recovered monotonic character error rate (CER)–budget frontiers across multilingual conditions. The dense FLEURS frontier in Tables 5 and 6 also exposes Pareto-efficient operating points and repeat-level latency outliers.

Table 7 adds the full-split multilingual stress run. We treat it as diagnostic evidence outside the headline results: the same threshold barely escalates outside Spanish, and Hindi/Thai have markedly heavier latency tails. Tables 8 and 9 separate language-specific target-30 quality from calibration cost. Budget calibration succeeds across all six languages and improves CER in every language-specific setting, while the latency view makes the heavier Hindi/Thai tails explicit. The full target-5/10/20/30/50/70 frontier in Table 10 confirms a monotonic mean CER recovery from 23.64 to 19.09 as escalation rises from 5% to 70%, with the absolute escalation-rate difference between target and realized rate below 0.3% for every non-degenerate target except Spanish target-5. The high-budget frontier in Table 11 extends the curve to 18.62 and 18.16 mean CER at 79.97% and 90.02% escalation, with the expected increase in 95th-percentile real-time factor (p95 RTF) from heavier strong-model use.

Table 1: Preliminary FLEURS English cascade results. The high-threshold runs completed all shards but did not escalate, indicating that the first hand-designed difficulty score requires calibration.

Run	Threshold	WER (%)↓	CER (%)↓	Escalation (%)↓	Examples
Wave 1	0.62	26.71	7.16	0.00	3643
Wave 2	0.50	26.71	7.16	0.00	3643
Calibrated 5%	0.212	26.67	7.13	5.02	3643
Calibrated wave 1	0.198	26.49	6.98	10.02	3643
Calibrated wave 2	0.186	26.46	6.96	20.01	3643
Calibrated 30%	0.180	26.31	6.84	29.98	3643
Calibrated 50%	0.170	26.19	6.81	50.01	3643
Calibrated 70%	0.162	25.95	6.63	70.00	3643
Low-threshold wave 1	0.15	25.90	6.57	91.68	3643
Low-threshold wave 2	0.10	25.89	6.57	99.97	3643

Table 2: Multilingual FLEURS character-error-rate matrix. Each cell reports CER (%)↓ for the indicated cascade condition; "Cheap/High" is the cheap high-threshold baseline and "Strong-ref" is the strong-recognizer reference. Separating quality from escalation keeps the language comparison readable.

Language	Cheap/ High CER (%)↓	Cal. 10% CER (%)↓	Cal. 30% CER (%)↓	Cal. 50% CER (%)↓	Cal. 70% CER (%)↓	Strong ref. CER (%)↓
Mandarin	59.07	57.97	56.10	54.48	53.01	50.84
Thai	25.13	22.95	20.31	18.35	16.31	14.44
Spanish	17.58	16.08	15.78	15.58	15.45	15.32
French	7.08	6.64	6.20	5.78	5.45	5.12
German	10.11	9.85	9.44	9.05	8.84	8.59
Hindi	28.13	25.06	20.92	17.81	15.37	12.35

Table 3: Multilingual FLEURS escalation matrix for the conditions in Table 2. Each cell reports escalation (%)↓; "Cheap/High" is the cheap high-threshold baseline and "Strong-ref" invokes the strong recognizer for every example. The target-calibrated columns show the intended budget.

Language	Cheap/ High Esc. (%)↓	Cal. 10% Esc. (%)↓	Cal. 30% Esc. (%)↓	Cal. 50% Esc. (%)↓	Cal. 70% Esc. (%)↓	Strong ref. Esc. (%)↓
Mandarin	0	10	30	50	70	100
Thai	0	10	30	50	70	100
Spanish	0	12	30	50	70	100
French	0	10	30	50	70	100
German	0	10	30	50	70	100
Hindi	0	10	30	50	70	100

Table 4: Difficulty-score saturation in multilingual FLEURS small-reference runs. The final column reports the actual escalation obtained when targeting 5% escalation by empirical thresholding.

Language	Examples	Unique scores	Max-score mass (%)↓	Actual at 5% (%)↓
Mandarin	4600	379	91.78	91.78
Thai	4062	2457	39.27	39.27
Spanish	4112	3631	0.02	5.01
French	4158	4158	0.02	5.00
German	4212	4212	0.02	5.01
Hindi	2777	2774	0.14	5.01

Table 5: FLEURS cascade quality frontier. Repeated target settings average repeat runs; efficiency reports the absolute CER reduction (%) per additional 0.01 p95 real-time factor relative to the 5% target. The companion table reports the latency variability.

Target	Runs	Esc. (%)↓	WER (%)↓	CER (%)↓	ΔCER (%)↓	Eff.↑
5	2	5.05	26.63	7.10	+0.00	–
10	2	11.57	26.45	7.00	-0.10	0.00
20	2	20.55	26.45	6.96	-0.14	0.00
30	2	30.00	26.30	6.84	-0.26	0.23
40	2	39.99	26.20	6.80	-0.30	0.24
50	1	50.07	26.07	6.72	-0.38	0.28
60	2	59.99	25.98	6.66	-0.45	0.29
70	1	70.05	25.95	6.63	-0.47	0.30
80	2	80.02	25.95	6.62	-0.48	0.27
90	2	90.01	25.91	6.59	-0.52	0.27

Table 6: FLEURS cascade latency variability for the targets in Table 5. The p95 real-time factor (RTF) and repeat spread are reported separately so the quality frontier remains readable.

Target	Runs	p95 RTF↓	Repeat Δp95↓
5	2	0.0214	0.0008
10	2	0.4248	0.7987
20	2	0.3148	0.5709
30	2	0.0327	0.0001
40	2	0.0339	0.0012
50	1	0.0352	–
60	2	0.0367	0.0009
70	1	0.0371	–
80	2	0.0393	0.0000
90	2	0.0403	0.0015

Table 7: FLEURS multilingual full-split stress baseline. The same small-to-large-v3-turbo cascade and threshold are used across languages; results are diagnostic evidence outside the headline results.

Language	N	WER (%)↓	CER (%)↓	Esc. (%)↓	p95 RTF↓
Chinese (Mandarin, Hans)	4600	100.00	59.07	0.00	0.0255
German	4212	47.13	10.07	0.00	0.0217
Spanish (LatAm)	4112	31.13	16.08	11.75	0.0216
French	4158	25.43	7.08	0.00	0.0224
Hindi	2777	61.46	28.23	0.07	0.3330
Thai	4062	123.45	24.89	0.20	0.0583

Table 8: FLEURS full-split language-calibrated target-30 routing quality. Values are percentages; lower character error rate (CER) is better.

Language	Base Esc. (%)↓	Cal. Esc. (%)↓	Base CER (%)↓	Cal. CER (%)↓	ΔCER (%)↓
Chinese (Mandarin, Hans)	0.00	30.02	59.07	56.10	-2.97
German	0.00	30.03	10.07	9.50	-0.58
Spanish (LatAm)	11.75	30.03	16.08	15.78	-0.30
French	0.00	30.04	7.08	6.20	-0.88
Hindi	0.07	29.20	28.23	21.19	-7.03
Thai	0.20	29.94	24.89	20.39	-4.50

Table 9: FLEURS language-calibrated target-30 routing cost. Lower escalation and p95 real-time factor (RTF) indicate lower resource use; the target is approximately 30% escalation per language.

Language	Cal. Esc. (%)↓	Cal. p95 RTF↓
Chinese (Mandarin, Hans)	30.02	0.0398
German	30.03	0.0332
Spanish (LatAm)	30.03	0.0329
French	30.04	0.0353
Hindi	29.20	0.2692
Thai	29.94	0.1292

Table 10: FLEURS language-calibrated routing frontier averaged across languages. Values are percentages except p95 RTF.

Target	Runs	Esc. (%)↓	|gap|↓	CER (%)↓	p95 RTF↓
5	6	6.08	1.20	23.64	0.0762
10	5	9.82	0.28	24.56	0.0926
20	6	19.95	0.11	22.24	0.0866
30	6	29.88	0.16	21.53	0.0899
50	6	49.92	0.08	20.17	0.0970
70	6	69.89	0.12	19.09	0.0991

Table 11: FLEURS high-budget language-calibrated routing frontier averaged across languages. Values are percentages except p95 RTF.

Target	Runs	Esc. (%)↓	|gap|↓	CER (%)↓	p95 RTF↓
80	6	79.97	0.04	18.62	0.2539
90	6	90.02	0.03	18.16	0.2608

2 ContextASR-Bench Policy Details

These tables support the main ContextASR-Bench result (Wang et al., 2025b). Table 13 separates model-size effects from contextual evidence. Tables 14 and 15 separate held-out context-gate quality from budget and latency cost. Table 17 gives the direct paired comparison of the selected pair rule against fixed p70. Tables 16 and 18 report additional held-out routing checks, while Tables 19, 20, and 21 keep harm, quality, and joint-policy cost behavior explicit.

The additional records include a strong-budget recovery sweep in Tables 22 and 23, full 15,000-example prompt-frontier checkpoints in Table 24, and an earlier repair-rerun ablation in Tables 25 and 26. Within that earlier rerun, the cheap-miss guard has the lowest CER (7.97%) at higher recorded p95 RTF (0.0728). Its pair-rule row (22.43% WER, 8.25% CER) is a separate saved decode from the selected comparison in main Table ?? (21.94% WER, 7.85% CER). The earlier run directories and paired predictions are unavailable for replay; these rows are descriptive within their own rerun and are not pooled with the main comparison.

Selector and budget contract. The fixed p70 reference uses the frozen p70 (70th-percentile) difficulty threshold and top-k entity sidecar. The train/dev pair rule enumerates duration quantiles $\{0.60, 0.65, 0.70, 0.75, 0.80, 0.85, 0.90\}$ and feature quantiles $\{0.25, 0.35, 0.45, 0.50, 0.60, 0.70, 0.80\}$, then selects a binary gate on development CER, WER, negative entity F1, prompt harm, and context rate. Its evaluated held-out gate is the disjunction chunk_seconds < 54.33 or prompt_chars_per_second > 2.9971811150506786; the thresholds are fixed before held-out decoding.

The joint-cost pair rule uses the same candidates with recorded weights 1 for WER, 1 for CER, -0.25 for entity F1, 0.05 for harm, 0.01 for context rate, and 4 for p95 RTF. The external record contains a separate four-fold cross-fit action trace, but its paired prediction shards are absent. It is retained only for aggregate action accounting; this appendix does not interpret its stored CER values as a new routing result.

Table 12 anchors the static endpoints used to interpret the policy settings. The fixed-context and no-context entries isolate the supplied-sidecar condition used by the prospective selector; the table does not establish a causal comparison of recognizer capacity.

Table 27 refreshes the original 512-example latency probe with matched accelerator shards. At the same 72.46% strong-model escalation, p70 entity context reduces WER/CER from 29.54%/10.97% to 23.99%/9.53%. Mean total latency changes only from 2.479 to 2.489 seconds, and p95 RTF remains effectively flat (0.1525 versus 0.1510). Table 28 repeats the latency check on all 15000 ContextASR examples using matched accelerator shards. The full-split refresh preserves the contextual-prompting gain (8.96% to 8.20% CER) and lowers p95 RTF from 0.0577 to 0.0476. The result is therefore a full-split, hardware-bounded check, not only a small-probe sanity result.

Table 12: Static and gated ContextASR baselines on the 15000-example setting. Small-only is derived from the cheap-pass transcripts of the matched no-context run; full-strong was decoded prospectively with 100% strong escalation. Quality values are percentages.

Policy	Strong (%)↓	Context (%)↓	WER (%)↓	CER (%)↓	Ent. F1↑
Small only	0.00	0.00	26.84	8.95	59.34
No-context cascade	66.65	0.00	26.53	8.96	66.41
Full strong	100.00	0.00	26.45	9.04	69.25
p70 context	66.65	64.63	22.27	8.27	84.00
Pair rule	66.65	60.69	22.24	8.09	83.09
Tree gate	66.64	61.33	22.13	8.05	83.31

Table 13: ContextASR-Bench scale-up at matched strong-model escalation. Prompt use is the percentage of examples receiving entity prompts during strong decoding.

Setting	Prompt use (%)↓	WER (%)↓	CER (%)↓	Entity recall↑	Entity F1↑
256 no context	0.00	29.51	10.81	51.65	68.12
256 p70 gate	70.31	23.71	8.86	75.80	86.23
512 no context	0.00	29.36	10.87	50.96	67.52
512 p70 gate	70.70	24.13	9.51	75.44	86.00
1024 no context	0.00	29.11	10.80	50.43	67.05
1024 p70 gate	69.14	25.04	10.04	74.03	85.07
2000 no context	0.00	28.67	10.48	50.40	67.02
2000 p70 gate	68.45	24.34	9.59	73.73	84.88
4000 no context	0.00	28.04	9.96	51.02	67.57
4000 p70 gate	68.67	23.74	9.29	73.94	85.02
8000 no context	0.00	26.94	9.29	49.98	66.65
8000 p70 gate	63.15	22.67	8.49	71.81	83.59
15000 no context	0.00	26.53	8.96	49.71	66.41
15000 p70 gate	64.63	22.27	8.27	72.42	84.00

Table 14: Prospective context-gate quality on the 15,000-example ContextASR setting. WER and CER are percentages; entity F1 is computed against the full manifest entity lists. Lower error rates and resource use are better, while higher entity F1 is better.

Run	Strong (%)↓	Context (%)↓	WER (%)↓	CER (%)↓	ΔWER (%)↓	ΔCER (%)↓	Ent. F1↑
No context	66.65	0.00	26.53	8.96	0.00	0.00	66.41
p70 context	66.65	64.63	22.27	8.27	-4.27	-0.69	84.00
duration cap	66.63	48.74	23.13	8.24	-3.41	-0.72	80.04
pair rule	66.65	60.69	22.24	8.09	-4.29	-0.87	83.09
tree gate	66.64	61.33	22.13	8.05	-4.40	-0.92	83.31

Table 15: Prospective context-gate budget and latency deltas on the same 15,000-example ContextASR setting. Lower error deltas, p95 wall-clock time, and p95 real-time factor (RTF) are better; the quality columns use the no-context run as the reference.

Run	ΔWER (%)↓	ΔCER (%)↓	p95 s↓	p95 RTF↓
No context	0.00	0.00	1.253	0.0255
p70 context	-4.27	-0.69	2.455	0.0504
duration cap	-3.41	-0.72	1.363	0.0260
pair rule	-4.29	-0.87	1.699	0.0368
tree gate	-4.40	-0.92	1.814	0.0376

Table 16: Paired bootstrap intervals on the disjoint ContextASR-Bench test split. Values are absolute differences (row minus no context) in the named metrics, with 2000 paired utterance resamples; WER and CER differences use %.

Run	ΔWER (%)↓	ΔCER (%)↓	ΔEntity F1↑	Prompt (%)↓
p70-context	-4.55 [-5.12, -3.94]	-0.83 [-1.24, -0.40]	+17.83 [+17.30, +18.35]	64.56 [63.07, 66.13]
Full-run-pair	-4.69 [-5.22, -4.14]	-1.14 [-1.49, -0.77]	+16.95 [+16.41, +17.49]	60.83 [59.25, 62.43]
Train-dev-pair	-4.73 [-5.26, -4.18]	-1.20 [-1.57, -0.82]	+16.81 [+16.27, +17.35]	60.27 [58.75, 61.87]
Joint-cost-pair	-3.91 [-4.38, -3.40]	-0.91 [-1.25, -0.59]	+14.86 [+14.31, +15.39]	52.88 [51.20, 54.43]

Table 17: Direct paired comparison of the train/dev pair rule against fixed p70 context on the shared 3,750-utterance held-out partition (2,000 paired utterance bootstrap resamples). Negative WER/CER changes are better; positive entity-F1 changes are better.

Comparison	ΔWER (%) ↓	ΔCER (%) ↓	ΔEntity F1 (%) ↑
Train/dev pair minus fixed p70	-0.17 [-0.57, +0.23]	-0.37 [-0.71, -0.03]	-1.01 [-1.19, -0.84]

Table 18: ContextASR policy ladder on the 15000-example p70 pair. Rows combine validated held-out split diagnostics to compare fixed prompting, learned gates, and oracle upper bounds under the same no-context/p70 run pair. Values are percentages over five repeated 50/50 splits.

Policy	Context (%)↓	WER (%)↓	CER (%)↓	Entity F1↑	Role
No context	0.00	26.61	8.96	66.37	Fixed baseline
p70 context	64.59	22.24	8.23	84.07	Fixed prompt rule
RF utility gate	61.07	22.25	8.08	83.31	Learned cost gate
GB latency gate	55.83	22.09	7.72	82.28	Tail-latency gate
RF harm gate	53.88	22.23	7.83	82.51	Learned harm gate
Oracle utility	58.58	20.53	6.81	83.24	Utility upper bound
Oracle no-harm	54.88	20.63	6.74	82.37	Harm upper bound

Table 19: Two-stage harm-gate diagnostic on the 15000-example p70 pair after adding duration, prompt-overlap, and no-context transcript-disagreement features. Harm is the fraction of selected context prompts whose word or character errors exceed the no-context decode. Values are held-out percentages over five repeated 50/50 splits.

Gate	Pen.	Context (%)↓	WER (%)↓	CER (%)↓	Entity F1↑	Harm (%)↓
p70 context	–	64.59 ± 0.14	22.24 ± 0.14	8.23 ± 0.08	84.07 ± 0.10	13.44 ± 0.27
Oracle	4.00	58.58 ± 0.08	20.53 ± 0.04	6.81 ± 0.02	83.24 ± 0.09	6.31 ± 0.25
Oracle no-harm	4.00	54.88 ± 0.20	20.63 ± 0.04	6.74 ± 0.02	82.37 ± 0.12	0.00 ± 0.00
RF harm	4.00	59.15 ± 1.25	22.06 ± 0.15	7.87 ± 0.10	83.13 ± 0.18	11.39 ± 0.32
RF harm	8.00	53.88 ± 1.60	22.23 ± 0.20	7.83 ± 0.11	82.51 ± 0.21	10.47 ± 0.42
GB harm	8.00	51.78 ± 2.07	22.33 ± 0.16	7.86 ± 0.13	82.22 ± 0.37	10.24 ± 0.52

Table 20: Quality and harm view of the joint context-policy diagnostic on the 512-example English FLEURS latency pair. The analysis reuses already decoded no-context and p70-context outputs. The oracle configuration is an upper bound under the stated penalties, while RF/GB are split-trained gates. Harm is the percentage of selected prompts whose WER or CER is worse than the no-context transcript.

Policy	**Context (%)↓**	**WER (%)↓**	**CER (%)↓**	**Harm (%)↓**	**Prov.↓**
No context	0.00 ± 0.00	29.71 ± 0.37	11.08 ± 0.31	0.00 ± 0.00	0.000 ± 0.000
p70 context	71.48 ± 1.28	24.19 ± 1.00	9.74 ± 0.65	13.78 ± 0.75	0.193 ± 0.041
oracle joint	59.06 ± 2.03	21.67 ± 0.29	7.29 ± 0.15	0.40 ± 0.33	0.151 ± 0.048
rf joint	60.78 ± 4.90	24.36 ± 1.02	9.36 ± 0.74	12.06 ± 1.46	0.188 ± 0.049
gb joint	60.86 ± 2.92	24.90 ± 1.01	9.86 ± 0.67	12.27 ± 1.25	0.164 ± 0.039

Table 21: Penalty and cost view of the joint context-policy diagnostic. The four λ columns are the fixed weights used by the joint utility; p95 RTF and Prov. report the corresponding latency and selected provenance-risk feature.

Policy	λ_c	λ_{lat}	λ_h	λ_p	**p95 RTF↓**	**Prov.↓**
No context	–	–	–	–	0.058 ± 0.011	0.000 ± 0.000
p70 context	–	–	–	–	0.059 ± 0.005	0.193 ± 0.041
oracle joint	0.01	4.00	0.05	0.01	0.024 ± 0.000	0.151 ± 0.048
rf joint	0.01	4.00	0.05	0.01	0.050 ± 0.016	0.188 ± 0.049
gb joint	0.01	4.00	0.05	0.01	0.055 ± 0.015	0.164 ± 0.039

Table 22: Quality view of the completed ContextASR-Bench 15k strong-budget recovery sweep. Rows vary the nominal context/strong-use budget while preserving matched no-context and contextual decode pairs. Quality values are percentages; lower WER is better and higher entity F1 is better.

Budget	**Strong (%)↓**	**Prompt (%)↓**	**NoCtx WER (%)↓**	**Ctx WER (%)↓**	**ΔWER (%)↓**	**Entity F1↑**
t10	10.01	10.01	26.78	26.16	-0.62	63.75
t30	30.06	30.06	26.70	24.59	-2.11	71.93
t50	50.04	50.04	26.68	23.45	-3.24	79.15
t70	66.65	64.63	26.53	22.27	-4.27	84.00

Table 23: CER view of the completed ContextASR-Bench 15k strong-budget recovery sweep. The matched no-context and contextual decodes use the same examples as Table 22; lower CER is better.

Budget	**NoCtx CER (%)↓**	**Ctx CER (%)↓**	**ΔCER (%)↓**
t10	8.94	8.87	-0.07
t30	8.94	8.53	-0.41
t50	9.00	8.51	-0.49
t70	8.96	8.27	-0.69

Table 24: Refreshed ContextASR-Bench 15000-example prompt frontier. Rows increase contextual prompting rate without changing the underlying no-context and contextual decode pair; quality metrics are percentages.

Run	Prompt (%)↓	WER (%)↓	CER (%)↓	ΔWER (%)↓	Entity recall↑	Entity F1↑
No context	0.00	26.53	8.96	–	49.71	66.41
p10	6.46	26.18	8.95	-0.35	52.03	68.45
p20	16.31	25.42	8.79	-1.11	55.51	71.39
p30	23.36	25.02	8.76	-1.51	57.98	73.40
p40	35.31	24.26	8.62	-2.28	62.19	76.69
p50	47.35	23.47	8.48	-3.06	66.42	79.82
p60	56.20	22.89	8.41	-3.64	69.47	81.98

Table 25: Earlier ContextASR-Bench repair-rerun ablation, saved aggregate only. These decodes differ from the selected main-paper comparison and are interpreted within this rerun. WER and CER are percentages; absolute differences use this table's no-context row as reference. Lower errors and higher entity F1 are better.

Run	WER (%)↓	CER (%)↓	ΔWER (%)↓	ΔCER (%)↓	Ent. F1↑
No-context	26.84	9.19	+0.00	+0.00	66.15
p70-context	22.40	8.36	-4.44	-0.83	83.95
pair-rule repair rerun	22.43	8.25	-4.41	-0.94	82.95
cheapmiss-guarded	22.13	7.97	-4.72	-1.22	82.81
duration60-guarded	22.37	8.14	-4.47	-1.05	82.57
duration60-cheapmiss	22.53	8.20	-4.31	-1.00	82.34
joint-cost	22.77	8.17	-4.07	-1.02	81.01

Table 26: Budget and latency saved for the earlier repair rerun in Table 25. Prompt use and p95 real-time factor (RTF) are resource measures; lower values are better. The recorded setting is not used to estimate latency for the main-paper comparison.

Run	Prompt (%)↓	p95 RTF↓
No-context	0.00	0.0396
p70-context	64.56	0.0590
pair-rule repair rerun	60.24	0.0549
cheapmiss-guarded	59.04	0.0728
duration60-guarded	58.91	0.0569
duration60-cheapmiss	57.47	0.0575
joint-cost	52.85	0.0508

Table 27: ContextASR-512 latency refresh on matched accelerator shards. Values are percentages except seconds and RTF.

Condition	Rows	Esc. (%)↓	CER (%)↓	Mean s↓	p95 s↓	p95 RTF↓
No context	512	72.46	10.97	2.479	8.389	0.1525
p70 entity context	512	72.46	9.53	2.489	8.592	0.1510

Table 28: ContextASR-15000 latency refresh on matched accelerator shards. Values are percentages except seconds and RTF.

Condition	Rows	Esc. (%)↓	CER (%)↓	Mean s↓	p95 s↓	p95 RTF↓
No context	15000	66.64	8.96	1.474	2.651	0.0577
p70 entity context	15000	66.65	8.20	0.865	2.244	0.0476

3 Provenance and External Controls

These controls answer likely reviewer questions about unsupported prompts, oracle-context leakage, external file-list integrity, and evaluation criteria without crowding the main paper. Tables 29, 30, 31, and 32 trace the source and coverage checks applied before scoring. Tables 33 and 34 then report the replayed gate and its paired deltas. The external results reported below use only the corresponding leakage and provenance controls.

3.1 Responsible Use and Failure-Mode Gates

AdaSpeechRAG is a routing and evidence-selection framework, not a replacement for human review in high-stakes transcription. A deployment candidate should be blocked unless it passes five checks: (i) every prompt phrase has non-oracle provenance; (ii) context-text leakage is below the paper-facing threshold; (iii) speaker/session groups are disjoint across train, development, and test splits; (iv) slice reports cover dialect, speaker, duration, overlap, and entity-heavy subsets when those labels are available; and (v) the selected policy improves the requested metric without hiding regressions in normalized WER/CER or entity recall. These gates explain why the Fluent Speech Commands (FSC) ontology condition is reported as a negative control. SmartGlasses and Speaker-Reasoner-style evaluation are not scored here because licensed data and complete STM hypotheses are unavailable (Gao et al., 2026; Lin et al., 2026).

Table 29: Quality view of the prompt-provenance ablation on HF-indexed LexiAdapt-FLEURS sidecars. Each entry is the absolute difference in the named metric (%), prompt set minus its matched no-context full-strong reference; lower WER/CER is better and higher entity recall is better.

Locale	**Prompt set**	**ΔWER (%)↓**	**ΔCER (%)↓**	**ΔEntity recall↑**
English	All audited	-0.10 [-0.28, +0.09]	-0.06 [-0.16, +0.04]	+16.49 [+9.61, +23.60]
English	Runtime filter	+0.01 [-0.04, +0.05]	-0.05 [-0.08, -0.02]	+16.49 [+9.61, +23.60]
Spanish	All audited	-3.34 [-4.04, -2.70]	-0.59 [-0.79, -0.39]	+6.25 [+0.00, +14.00]
Spanish	Target-only	-0.42 [-0.86, +0.05]	+0.04 [-0.17, +0.40]	+6.25 [+0.00, +14.00]
Spanish	Runtime filter	-0.51 [-0.91, -0.17]	-0.06 [-0.17, +0.09]	+6.25 [+0.00, +14.00]
Spanish	Distractor-only	-3.13 [-3.78, -2.54]	-0.55 [-0.74, -0.35]	+0.00 [+0.00, +0.00]
Hindi	All audited	+12.52 [+9.02, +16.11]	+16.02 [+12.14, +20.06]	+28.75 [+15.91, +41.03]
Hindi	Target-only	-1.18 [-2.09, -0.48]	-0.45 [-1.43, +0.31]	+35.00 [+23.53, +46.07]
Hindi	Runtime filter	-1.03 [-1.85, -0.34]	-0.38 [-1.35, +0.40]	+35.80 [+23.75, +46.59]
Hindi	Distractor-only	+17.35 [+13.63, +21.26]	+21.93 [+17.71, +26.37]	+0.00 [+0.00, +0.00]

Table 35 examines an earlier group-disjoint FSC split with 4,371 paired test utterances. Fixed ontology context reduces raw WER from 29.22% to 25.01%, an absolute WER difference of -4.21% from unrounded scores. Normalized WER rises from 7.76% to 8.42%, an absolute difference of +0.67% from unrounded scores. Table 36 shows that 943 utterances improve in raw word error but tie after normalization; only 47 improve in both.

The earlier split used 21,454 training, 4,218 development, and 4,371 test utterances. On its development split, no context had 5.39% normalized WER and 2.86% normalized CER, and a normalized-error pair-rule search selected no context. Table 35 reports only the earlier fixed-context test comparison, not a gain from that selected rule.

Table 30: Coverage view of the prompt-provenance ablation. Prompt use and mean phrase counts describe the same sidecars as Table 29; distractor-only settings contain no reference-supported target entities, so their entity-recall deltas are zero by construction.

Locale	Prompt set	Prompt used (%)↓	Mean phrases↓
English	All audited	53.52	12.00
English	Runtime filter	8.40	0.19
Spanish	All audited	100.00	12.00
Spanish	Target-only	16.02	0.25
Spanish	Runtime filter	16.02	0.25
Spanish	Distractor-only	100.00	11.75
Hindi	All audited	100.00	12.00
Hindi	Target-only	23.83	0.31
Hindi	Runtime filter	23.83	0.31
Hindi	Distractor-only	100.00	11.69

Table 31: Source-evidence support audit for the HF-indexed LexiAdapt-FLEURS sidecars. The audit compares each sidecar phrase against the source manifest's nested context_phrase_evidence.

Locale	Prompt set	Prompted (%)↓	Mean phrases↓	Source supported phrases↑	Unsupported phrases↓
Spanish	All audited	100.00	12.00	64	3008
Spanish	Target-only	16.02	0.25	64	0
Spanish	Distractor-only	100.00	11.75	0	3008
Hindi	All audited	100.00	12.00	80	2992
Hindi	Target-only	23.83	0.31	80	0
Hindi	Distractor-only	100.00	11.69	0	2992

The main-paper FSC table uses a separate, later group-disjoint split with 20,994 training, 3,803 development, and 5,246 test utterances. On its development split, fixed context had 6.38% normalized WER and 3.17% normalized CER, against 6.45% and 3.17% for no context; the stored selection chose fixed context. The later training split reversed that ordering: fixed context reached 7.78% normalized WER and 4.62% normalized CER, against 7.16% and 4.08% for no context. On the later held-out test split, fixed context likewise worsened normalized WER/CER to 4.76%/2.43% from 4.54%/2.21% for no context, an absolute difference of +0.22% in each displayed metric. The main-paper "dev-selected" condition is the same held-out condition as fixed ontology context: development selected that action, and its test metrics reuse the fixed-context decode. It is not a separate held-out condition and is not a separately fitted or decoded policy; it confers no selection gain. The earlier deltas and utterance categories do not estimate the later test result. The split manifests establish their own partition assignments, but the original example IDs needed to quantify overlap between the earlier and later assignments are unavailable; we therefore make no independent-replication or filtered-subset claim. The later aggregate's exact paired predictions are unavailable for replay or an interval. Across these distinct FSC protocols, a plausible ontology prompt improves raw WER yet can worsen normalized recognition; neither protocol establishes natural-context transfer.

Table 32: Source-evidence status audit for the HF-indexed LexiAdapt-FLEURS sidecars. Unknown phrases are zero for every prompt set, and mixed cases are absent after applying the source-evidence partition.

Locale	Prompt set	Unknown phrases↓	Mixed cases↓
Spanish	All audited	0	41
Spanish	Target-only	0	0
Spanish	Distractor-only	0	0
Hindi	All audited	0	61
Hindi	Target-only	0	0
Hindi	Distractor-only	0	0

Table 33: Evidence-count view of a conservative source-evidence provenance gate over decoded HF-indexed LexiAdapt-FLEURS sidecars. The gate accepts only sidecars with zero unknown phrases, zero unsupported phrases, and at least one reference-supported phrase.

Locale	Prompt set	Gate	Ref.↑	Unsup.↓	Unknown↓
Spanish	All audited	reject	64	3008	0
Spanish	Target-only	accept	64	0	0
Spanish	Distractor-only	reject	0	3008	0
Hindi	All audited	reject	80	2992	0
Hindi	Target-only	accept	80	0	0
Hindi	Distractor-only	reject	0	2992	0

Table 34: Quality-delta view of the source-evidence provenance-gate replay. The deltas use the same sidecars and gate decisions as Table 33; lower WER/CER is better and higher entity recall is better.

Locale	Prompt set	ΔWER (%)↓	ΔCER (%)↓	ΔRecall↑
Spanish	All audited	-3.34 [-4.04, -2.70]	-0.59 [-0.79, -0.39]	+6.25 [+0.00, +14.00]
Spanish	Target-only	-0.42 [-0.86, +0.05]	+0.04 [-0.17, +0.40]	+6.25 [+0.00, +14.00]
Spanish	Distractor-only	-3.13 [-3.78, -2.54]	-0.55 [-0.74, -0.35]	+0.00 [+0.00, +0.00]
Hindi	All audited	+12.52 [+9.02, +16.11]	+16.02 [+12.14, +20.06]	+28.75 [+15.91, +41.03]
Hindi	Target-only	-1.18 [-2.09, -0.48]	-0.45 [-1.43, +0.31]	+35.00 [+23.53, +46.07]
Hindi	Distractor-only	+17.35 [+13.63, +21.26]	+21.93 [+17.71, +26.37]	+0.00 [+0.00, +0.00]

Table 35: Earlier FSC public-ontology test split (N = 4,371), distinct from the later N = 5,246 test split in the main paper. Differences are fixed ontology context minus no context in the named error metric (%); negative values favor context. The fixed prompt improves raw scoring but worsens normalized scoring on this earlier split. These values do not measure the later split's dev-selected policy.

Evidence	N	Raw WER (%) Δ ↓	Raw CER (%) Δ ↓	Norm WER (%) Δ ↓	Norm CER (%) Δ ↓
Fixed ontology context vs. no context	4,371	-4.21	-0.36	+0.67	+0.57

Table 36: Per-utterance categories for the earlier $N = 4{,}371$ FSC comparison in Table 35. Rates use paired utterances as the denominator; categories describe word-error behavior, not the later test split or its policy selection.

Category	**Count**	**Rate (%)**	**Interpretation**
Raw improves, normalized improves	47	1.08	Context helps lexical content after normalization.
Raw improves, normalized ties	943	21.57	Raw gain is mostly casing, punctuation, or formatting.
Raw improves, normalized regresses	2	0.05	Raw metric hides normalized lexical harm.
Raw regresses or ties, normalized regresses	42	0.96	Context introduces normalized errors.
All other ties/regressions	3337	76.35	No normalized benefit from ontology prompting.

4 Archived External Action Accounting

Table 37 contains the only retained external display. It records the Wenet/WSC cross-fit aggregate and the action-rate fields checked against the stored per-decision trace (Wang et al., 2026a; Dai et al., 2025). The paired prediction shards and source-session manifests are unavailable, so its stored CER fields do not provide a regenerated decoder comparison or transfer result. Earlier sweeps, cross-corpus transfers, and tier-selection records are excluded because their matched prediction-level inputs are absent.

Table 37: Archived external action-accounting record under utterance-key modulo cross-fit. Each bucket was held out once, the preceding bucket selected its threshold, and the remaining buckets supplied training evidence. Cheap and strong denote the small and large-v3-turbo recognizers. CER, intervals, and gaps are stored aggregate record fields; they are not regenerated decoder results because paired prediction shards are unavailable. Strong-audio is the share of recorded chunk seconds sent to strong, a duration-weighted invocation proxy, not measured wall-clock cost. Source-session records and natural-context inputs are also unavailable. This table supports only consistency checks between the stored trace and aggregate action counts; it does not support routing-performance, natural-context, or session-disjoint transfer findings.

Policy	Esc. / strong-audio (% / %) ↓	Raw CER (%) ↓	ΔCER (%) vs. cheap ↓ 95% CI	Gap to strong CER (%) ↓
Wenet-Wu test ($N = 757$)				
Cheap	0.00 / 0.00	85.30	+0.00 [+0.00,+0.00]	5.22
Strong	100.00 / 100.00	80.09	-5.22 [-6.32,-4.08]	0.00
Diff. p70	69.48 / 72.96	82.00	-3.30 [-4.31,-2.25]	1.92
Learned p70	68.56 / 75.15	80.76	-4.54 [-5.53,-3.52]	0.67 [0.14,1.18]
WSC-Eval full ($N = 8{,}373$)				
Cheap	0.00 / 0.00	59.34	+0.00 [+0.00,+0.00]	12.07
Strong	100.00 / 100.00	47.27	-12.07 [-12.74,-11.36]	0.00
Diff. p70	70.14 / 72.82	50.29	-9.05 [-9.69,-8.36]	3.02
Learned p70	69.40 / 83.05	47.60	-11.73 [-12.36,-11.07]	0.33 [0.08,0.57]

References

L. Li, Y. Li, H. Xue, J. Liu, S. Fang, K. Wang, and L. Xie, "Delayed-KD: Delayed Knowledge Distillation based CTC for Low-Latency Streaming ASR," in Proc. Interspeech, 2025.

Z. Lin, K. Huang, W. Ren, L. Yang, and L. Xie, "Contextualized Automatic Speech Recognition with Dynamic Vocabulary Prediction and Activation," in Proc. Interspeech, 2025.

B. Mu, Y. Shao, K. Wei, D. Yu, and L. Xie, "Efficient Scaling for LLM-based ASR," in Proc. ASRU, 2025.

B. Mu, K. Wei, Q. Shao, Y. Xu, and L. Xie, "HDMoLE: Mixture of LoRA Experts with Hierarchical Routing and Dynamic Thresholds for Fine-Tuning LLM-based ASR Models," in Proc. ICASSP, 2025.

B. Mu, H. Liu, H. Xue, K. Wei, and L. Xie, "Hearing More with Less: Multi-Modal Retrieval-and-Selection Augmented Conversational LLM-Based ASR," arXiv:2508.01166, 2025.

B. Mu, X. Shi, X. Wang, H. Liu, J. Xu, and L. Xie, "LLM-ForcedAligner: A Non-Autoregressive and Accurate LLM-Based Forced Aligner for Multilingual and Long-Form Speech," arXiv:2601.18220, 2026.

W. Tian, B. Mu, G. Ma, X. Geng, Z. Zhao, and L. Xie, "dLLM-ASR: A Faster Diffusion LLM-based Framework for Speech Recognition," arXiv:2601.17902, 2026.

H. Wang, X. Wan, N. Zheng, K. Liu, H. Zhou, G. Li, and L. Xie, "CAMEL: Cross-Attention Enhanced Mixture-of-Experts and Language Bias for Code-Switching Speech Recognition," in Proc. ICASSP, 2025.

H. Wang, L. Ma, D. Guo, X. Wang, L. Xie, J. Xu, and J. Lin, "ContextASR-Bench: A Massive Contextual Speech Recognition Benchmark," arXiv:2507.05727, 2025.

H. Xue, Y. Tang, J. Zhang, X. Geng, and L. Xie, "Selective Invocation for Multilingual ASR: A Cost-effective Approach Adapting to Speech Recognition Difficulty," in Proc. Interspeech, 2025.

D. Gao et al., "The SLT 2026 SmartGlasses Challenge: Benchmarking Egocentric Multi-Talker Speech Recognition and Understanding with Audio-Language Models," arXiv:2608.12034, 2026.

Z. Lin et al., "Speaker-Reasoner: Scaling Interaction Turns and Reasoning Patterns for Timestamped Speaker-Attributed ASR," arXiv:2604.03074, 2026.

C. Wang et al., "WenetSpeech-Wu: Datasets, Benchmarks, and Models for a Unified Chinese Wu Dialect Speech Processing Ecosystem," arXiv:2601.11027, 2026.

Y. Dai et al., "WenetSpeech-Chuan: A Large-Scale Sichuanese Corpus with Rich Annotation for Dialectal Speech Processing," arXiv:2509.18004, 2025.

A. Conneau et al., "FLEURS: Few-shot Learning Evaluation of Universal Representations of Speech," in Proc. SLT, 2023.