Ontology Memory-Augmented ASR Correction for Long Text-Speech Interleaved Conversations

Anonymous ACL submission

Abstract

Automatic speech recognition (ASR) correction has traditionally focused on isolated utterances or short local contexts. However, as text and speech become increasingly interleaved in long interactions, ASR correction requires conversation-level contextual evidence. Existing ASR correction methods often rely on the current hypothesis or concatenate raw dialogue history. In such contexts, sparse correction evidence can be difficult to locate amid redundancy and noise. Addressing these challenges, we propose an ontology memory-augmented ASR correction framework for long text-speech interleaved conversations. The framework organizes preceding interaction history into a dynamically updatable ontology memory, where entities, terminology, surface variants, potential ASR confusions, and semantic relations are stored as retrievable nodes for context-grounded correction. To evaluate this setting, we construct RAMC-Corr, a dataset derived from MAGIC-RAMC for long-range ASR correction with grounded context. Experiments on RAMC-Corr show that our method improves over direct correction in 9 out of 10 paired backbone-setting combinations and encourages more selective and evidence-grounded corrections for context-dependent ASR errors.

1 Introduction

Automatic speech recognition (ASR) serves as a fundamental component in a wide range of real-world applications, including voice assistants, meeting transcription, medical consultation, instant messaging, and intelligent office systems (Kim et al., 2025; Billings and McDonnell, 2025; HangChen et al., 2025). With the rapid development of speech interaction, large language models, and multimodal assistants, ASR outputs are no longer used merely as isolated transcriptions of short voice commands. Instead, they are increasingly embedded in continuous human-computer interaction, where text and speech appear alternately and jointly constitute the evolving interaction state of the current conversation (Zhang et al., 2023a; Rubenstein et al., 2023). Before a new speech segment arrives, the system may have already accumulated substantial preceding context, such as user-typed messages, confirmed dialogue history, meeting agendas, medical descriptions, or task instructions. Such contextual information can provide valuable evidence for recognizing domain-specific entities, long-tail terminology, and semantically dependent expressions in subsequent speech segments, but how to effectively exploit it for ASR correction remains underexplored.

Figure 1: An example of context-grounded ASR correction in a long-range Chinese text-speech interleaved conversation. The original Chinese utterances are shown with English translations for readability. Earlier context in the same conversation provides evidence for correcting later ASR errors involving entities and domain-specific terms.

Figure 1 shows a representative context-grounded ASR correction case from our Chinese dataset. The grounded context introduces dialogue-specific entities and terms, including “三体”, “云天明”, “程心”, “罗辑”, and “执剑人”, which later serve as evidence for correcting spoken ASR hypotheses. For example, the ASR system misrecognizes the domain-specific term “执剑人” as the phonetically plausible but contextually incorrect

"指针人". This error is difficult to resolve from the local hypothesis alone, but can be corrected by grounding the correction in earlier conversational context.

These observations highlight the need for ASR correction methods that can exploit contextual evidence accumulated during an ongoing interaction. Existing ASR correction methods typically take the current ASR hypothesis as input and use language models to rewrite erroneous transcriptions (Yang et al., 2023; Chen et al., 2023). Some studies further incorporate external vocabularies, entity lists, candidate phrases, or domain knowledge to improve the correction of named entities and rare words (Ghosh et al., 2025; Im et al., 2025). However, these resources are usually static, whereas useful correction evidence in long text-speech interleaved conversations is often introduced dynamically by the interaction itself. Simply concatenating historical context with the current ASR hypothesis is also insufficient, since relevant evidence becomes sparse and difficult to locate as the interaction grows longer. Redundant turns, irrelevant information, and noisy ASR outputs can further obscure the contextual cues needed for correction. These limitations motivate a structured memory mechanism that transforms preceding interaction context into reusable correction evidence.

In this work, we study *context-grounded ASR correction* for long-range text-speech interleaved conversations. The task requires the system to correct subsequent ASR transcriptions using causally accessible interaction history. Different from utterance-level correction, the key challenge lies in how to organize, retrieve, and reuse previously established contextual knowledge, such as entities, aliases, domain concepts, and semantic constraints.

To address this challenge, we propose an ontology memory-augmented ASR correction framework. The framework extracts correction-relevant knowledge from preceding interaction history and organizes it into a conversation-level ontology memory. When a new ASR hypothesis arrives, the correction model retrieves relevant evidence from the ontology memory for context-grounded correction. The memory is also dynamically updated as the interaction proceeds, enabling newly introduced concepts and terminology to support subsequent correction.

To evaluate this setting, we construct RAMC-Corr, a context-grounded ASR correction dataset based on the MagicData-RAMC speech chat corpus (Yang et al., 2022). RAMC-Corr evaluates whether a system can use causally accessible grounded context to correct later ASR hypotheses in long-range interactive scenarios. Experiments on RAMC-Corr show that our method improves over direct correction in 9 out of 10 paired settings, demonstrating the effectiveness of explicit memory for context-grounded ASR correction.

In summary, our main contributions are as follows:

- We study context-grounded ASR correction for long-range text-speech interleaved conversations.
- We propose an ontology memory-augmented correction framework that organizes conversation-level contextual knowledge into a retrievable and dynamically updatable structured memory.
- We construct RAMC-Corr, a MagicData-RAMC-based dataset for evaluating long-range context-grounded ASR correction in text-speech interleaved conversations.
- Experiments across multiple LLM backbones show consistent gains over direct correction.

We will publicly release the code and dataset upon publication.

2 RAMC-Corr Dataset Construction

We build RAMC-Corr, a new context-grounded ASR correction benchmark derived from MagicData-RAMC (Yang et al., 2022). It is designed to simulate long-range text-speech interleaved dialogue scenarios, where subsequent ASR hypotheses are corrected using causally accessible reliable textual history. MagicData-RAMC is a Chinese multi-turn conversational speech corpus collected from real interaction scenarios, containing long temporally ordered conversations with speech segments and corresponding human transcriptions. This structure makes it suitable for constructing long-range context-grounded ASR correction examples.

For each dialogue, we automatically select a boundary based on the human transcriptions. The contiguous transcription segments from the beginning of the dialogue to the boundary are used as the grounded context, which serves as causally accessible textual evidence for subsequent ASR correction. The segments after the boundary form the online target region, where text segments are directly observed and speech segments are represented by

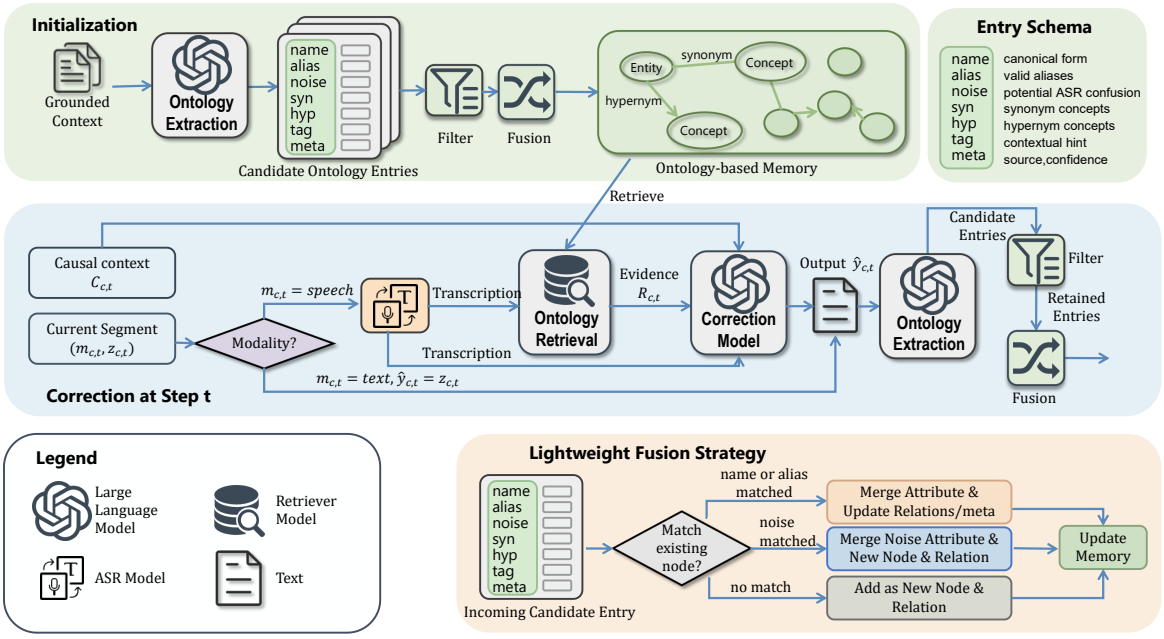

Figure 2: Overall architecture of the proposed ontology memory-augmented context-grounded ASR correction framework.

Split	#Conv.	#Samples	Boundary Ratio	Context Sim.
Train	268	268	0.076	0.654
Dev	17	17	0.071	0.638
Test	43	43	0.117	0.635
All	328	328	0.081	0.650

Table 1: Statistics of the RAMC-Corr dataset across training, development, and test splits. #Conv is the number of conversations, #Samples is the number of correction conversations, and Context Sim. is the contextual similarity used for boundary selection.

ASR hypotheses to be corrected. The corresponding human transcriptions of these speech segments are used as references for evaluation.

To ensure that the grounded context provides informative guidance for later correction, we require semantic continuity across the selected boundary. Specifically, we select candidate boundaries according to the contextual similarity between the grounded context and the initial part of the online target region, and filter out dialogues with weak cross-boundary semantic relatedness. For each valid dialogue, we generate at most one sample to avoid scattered or highly overlapping correction instances from the same conversation. The dataset statistics are summarized in Table 1, and the detailed construction procedure is presented in Appendix A.1.

3 Method

A direct strategy is to feed the ASR hypothesis together with accessible dialogue history into a correction model. However, this requires the model to locate useful entities, terminology, and semantic cues from an increasingly long and noisy context.

To address this issue, we propose an **ontology memory-augmented context-grounded ASR correction framework**, as illustrated in Fig. 2. Instead of relying solely on raw dialogue history, the framework maintains a conversation-level ontology memory that incrementally organizes reusable entities, terminology, domain concepts, surface variants, and semantic relations. The memory is first initialized from the grounded context and is then updated online after each processed segment.

During online processing, text segments are treated as reliable inputs and directly forwarded as outputs, while also being used to enrich the ontology memory. For each speech segment, the base ASR hypothesis is corrected by a correction model conditioned on the causal context and relevant ontology evidence retrieved from the current memory. After the output is produced, candidate ontology entries are extracted, filtered, and incorporated into the memory through a lightweight fusion strategy.

3.1 Task Definition

Given the c-th conversation sample, we consider a grounded context $\mathcal{G}_c$ followed by a temporally or-

dered processing sequence $\mathcal{X}_c$. The grounded context contains information available before online processing begins, while the processing sequence consists of text and speech segments:

$$\mathcal{X}_c = \{(m_{c,1}, z_{c,1}), \ldots, (m_{c,T_c}, z_{c,T_c})\}, \quad (1)$$

where $m_{c,t} \in \{\text{text}, \text{speech}\}$ denotes the modality of the t-th segment and $z_{c,t}$ denotes its content. Text segments are regarded as reliable inputs, whereas speech segments are transcribed by a base ASR system into hypotheses $a_{c,t}$ and serve as the targets of correction.

Under the streaming setting, when processing step t, the system can only access the grounded context and previous outputs:

$$C_{c,t} = (\mathcal{G}_c, \{\hat{y}_{c,i} \mid 1 \leq i < t\}). \quad (2)$$

This causal constraint prevents the system from using future segments.

The output at each step is defined as:

$$\hat{y}_{c,t} = \begin{cases} z_{c,t}, & m_{c,t} = \text{text}, \\ F(a_{c,t}, C_{c,t}), & m_{c,t} = \text{speech}, \end{cases} \quad (3)$$

where $F(\cdot)$ denotes a correction function that refines the ASR hypothesis using only causally accessible information.

3.2 Ontology Working Memory

We maintain a conversation-level ontology working memory to transform long-range interaction history into reusable correction evidence. Rather than a predefined static knowledge base, it is dynamically constructed within each conversation to store reusable entities, terminology, surface variants, and semantic relations.

We denote the memory before time step t in conversation c as $\mathcal{M}_{c,t}$, which consists of an ontology node set $\mathcal{V}_{c,t}$ and a semantic relation set $\mathcal{E}_{c,t}$. Each node represents a reusable entity, item, or domain concept, together with its aliases, potential ASR noise forms, contextual tags, and metadata. Relations encode semantic associations between concepts, such as synonym and hypernym relations.

3.2.1 Candidate Ontology Entries

Given causally accessible textual content, including the grounded context, reliable text segments, and corrected speech outputs, the ontology extractor produces a set of candidate ontology entries. Each entry follows a lightweight schema with seven fields: `name`, `alias`, `noise`, `syn`, `hyp`, `tag`, and `meta`. As shown in Figure 2, the schema records the canonical concept, aliases, potential ASR noise forms, synonym and hypernym links, contextual tags, and auxiliary metadata. Candidate entries serve as the basic unit for memory writing. During this process, the central concept and its variants are converted into ontology node attributes, while semantic associations such as synonymy and hypernymy are written into the relation set.

3.2.2 Initialization and Lightweight Fusion

Before online correction begins, we initialize the ontology working memory from the grounded context. We view the grounded context $\mathcal{G}_c$ as an ordered sequence of context segments $\{g_{c,\ell}\}_{\ell=1}^{L_c}$. Starting from an empty memory, the system processes these segments in order, extracts candidate ontology entries from each segment, and writes them into memory using the same update operation introduced above. The resulting memory is used as the initial online memory $\mathcal{M}_{c,1}$.

Incoming entries are compared with existing nodes using the `name`, `alias`, and `noise` fields. Matches on `name` or `alias` are treated as concept-level matches and are merged into the corresponding nodes. Matches supported only by the `noise` field are used to expand ASR-confusion evidence, but do not by themselves prevent the incoming entry from being added as a new node. For matched entries, node attributes and associated metadata are updated, while synonym and hypernym associations are incorporated into the relation set. Entries without concept-level matches are added as new nodes. When duplicated relations are observed, the system updates their metadata, such as occurrence count and confidence, instead of inserting redundant edges. The same fusion strategy is reused during online processing.

3.3 Ontology Evidence Retrieval

Directly conditioning ASR correction on the entire interaction history is inefficient and prone to entity ambiguity. We therefore retrieve structured ontology evidence from the current working memory, allowing the correction model to access localized contextual cues without searching through raw long-range history.

Because only speech segments require correction, ontology retrieval is performed only when the current segment is speech. Given the ASR hypothesis $a_{c,t}$, causal context $C_{c,t}$, and current ontology memory $\mathcal{M}_{c,t}$, the retriever obtains relevant ontol-

ogy evidence:

$$R_{c,t} = \mathrm{Retrieve}_{\eta}(a_{c,t}, C_{c,t}, \mathcal{M}_{c,t}), \quad m_{c,t} = \mathrm{speech}. \tag{4}$$

The retrieved evidence $R_{c,t}$ consists of matched ontology nodes together with their stored attributes and associated relations, such as aliases, potential ASR confusion forms, contextual tags, and synonym or hypernym associations. The retriever $\mathrm{Retrieve}_{\eta}$ is implementation-agnostic and can be instantiated with lexical matching, semantic retrieval, vector retrieval, or hybrid strategies.

3.4 Ontology-Augmented ASR Correction and Memory Update

ASR Correction. For a speech segment, the correction model generates the final transcription conditioned on the ASR hypothesis, causal context, and retrieved ontology evidence. We instantiate the correction function F in Eq. (3) as a parameterized model f_{θ}:

$$\hat{y}_{c,t} = f_{\theta}\left(a_{c,t}, C_{c,t}, R_{c,t}\right), \quad m_{c,t} = \mathrm{speech}. \tag{5}$$

The retrieved ontology evidence provides localized concept constraints and semantic cues that help resolve entity ambiguity and ASR confusions. For text segments, no correction is applied, and the reliable text is directly used as the output:

$$\hat{y}_{c,t} = z_{c,t}, \quad m_{c,t} = \mathrm{text}. \tag{6}$$

Memory Update. After the output $\hat{y}_{c,t}$ is determined, both corrected speech and reliable text are used as sources for memory enrichment. The ontology extractor obtains candidate entries from the current output:

$$\mathcal{B}_{c,t} = E_{\phi}\left(\hat{y}_{c,t}\right). \tag{7}$$

Unsupported, unstable, or non-reusable entries are filtered out, yielding the retained subset $\mathcal{B}^{+}_{c,t}$. The retained entries are then incorporated into the ontology memory using the lightweight fusion strategy described above:

$$\mathcal{M}_{c,t+1} = \mathrm{Update}\left(\mathcal{M}_{c,t}, \mathcal{B}^{+}_{c,t}\right). \tag{8}$$

If no valid entry is retained, the memory remains unchanged.

4 Experiments

4.1 Settings

Evaluation Metrics. We report character error rate (CER) in two forms: corpus CER (C-CER) and macro CER (M-CER), along with relative CER reduction (RelCER). C-CER aggregates edit distances over the whole set, while M-CER averages CER over individual correction instances. CER is computed with jiwer[1] after NFKC normalization, lowercasing, and removing punctuation and whitespace. RelCER is computed against the original ASR hypotheses:

$$\mathrm{RelCER} = \frac{\text{C-CER}_{\mathrm{Raw}} - \text{C-CER}_{\mathrm{Corrected}}}{\text{C-CER}_{\mathrm{Raw}}} \times 100.$$

Lower C-CER/M-CER and higher RelCER indicate better correction quality. All values are percentages.

Model Details. We use `inf-retriever-v1` to retrieve ontology entries from working memory. The ontology extractor and correction model use the same backbone LLM unless otherwise specified. For correction, we evaluate Qwen2.5-7B/14B/72B-Instruct and Qwen3.5-4B/9B. We also evaluate Gemma-4-26B-128K as a Full-History baseline, where the complete available history is directly provided without explicit working memory construction. Fro asr model, we use Qwen2-Audio-7B-Instruct. For readability, we omit the "Instruct" suffix in tables. All LLM backbones are evaluated in a training-free inference-only setting. We use vLLM (Kwon et al., 2023) for efficient model inference, and report all results from a single run unless otherwise specified.

Computational Setup. All experiments are inference-only and do not involve model training. Experiments were conducted on NVIDIA H100 SXM5 80GB GPUs.

Hyperparameters. Unless otherwise specified, we use 5 previous utterances as local context and retrieve the top 10 ontology entries from the working memory. For model inference, we set the temperature to 0.0, top-p to 1.0, and the maximum number of generated tokens to 2048. In the few-shot setting, both ontology information extraction and correction use 3-shot demonstrations as the default prompting hyperparameter. The demonstrations are manually selected from the development set and fixed across test examples within the same experimental setting. By default, all segments after the boundary are provided as speech inputs, with no text inputs given for the target region. In the ablation studies, we randomly replace a specified proportion of target-region speech inputs with their

[1] https://github.com/jitsi/jiwer

Model	Setting	Method	C-CER ↓	M-CER ↓	RelCER ↑
Raw ASR	–	–	27.94	55.44	0.00
Gemma-4-26B	ZS	Full-History	27.94	55.44	+0.50
	FS	Full-History	27.96	55.44	0.00
Qwen2.5-7B	ZS	Direct	36.40	73.01	-30.27
		Ours	**31.76** (+12.7)	58.33 (+20.1)	-13.67
	FS	Direct	35.50	67.59	-27.05
		Ours	33.00 (+7.0)	59.34 (+12.2)	-18.10
Qwen2.5-14B	ZS	Direct	35.66	84.57	-27.64
		Ours	**29.04** (+18.6)	55.35 (+34.6)	-3.95
	FS	Direct	34.60	77.40	-23.85
		Ours	32.43 (+6.3)	66.30 (+14.3)	-16.09
Qwen2.5-72B	ZS	Direct	29.82	57.19	-6.73
		Ours	**29.16** (+2.2)	55.62 (+2.7)	-4.36
	FS	Direct	30.38	58.31	-8.73
		Ours	30.06 (+1.1)	57.85 (+0.8)	-7.58
Qwen3.5-4B	ZS	Direct	27.61	55.17	+1.19
		Ours	27.36 (+0.9)	54.94 (+0.4)	+2.08
	FS	Direct	29.16	57.60	-4.36
		Ours	**27.32** (+6.3)	54.97 (+4.6)	+2.20
Qwen3.5-9B	ZS	Direct	27.47	55.29	+1.68
		Ours	**27.26** (+0.8)	55.17 (+0.2)	+2.42
	FS	Direct	28.02	55.85	-0.30
		Ours	28.89 (-3.1)	58.05 (-3.9)	-3.41

Table 2: Main results on the RAMC-Corr dataset. All values are percentages, and ZS/FS denote zero-shot/few-shot prompting. Values in parentheses indicate the relative reduction over Direct for C-CER and M-CER under the same backbone and prompting setting. Bold numbers indicate the lower C-CER between Direct and Ours.

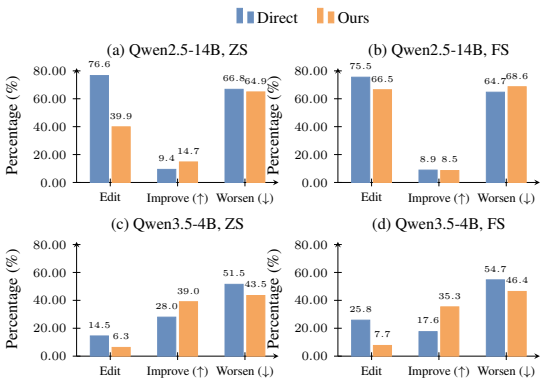

Figure 3: Edit behavior comparison under zero-shot and few-shot settings. Edit, Improve, and Worsen denote Edit Rate, Improve@Edit, and Worsen@Edit, respectively. ZS/FS denote zero-shot/few-shot prompting.

transcript form according to the predefined text-input ratio. We further analyze the effect of the number of in-context examples in Appendix A.2, and provide the prompt templates in Appendix B.

4.2 Main Results

Table 2 reports the main correction results on the RAMC-Corr dataset. We compare Direct correction, which feeds ASR hypotheses and recent corrected conversational context directly to the backbone model, with our memory-augmented correction method, which uses an explicit working memory constructed from dialogue history. We also include a Full-History baseline using Gemma-4-26B-128K, where the complete available conversation history is provided to the model without explicit memory construction. Overall, our method improves C-CER over Direct in 9 out of 10 paired backbone-setting combinations, indicating that explicit working memory provides useful correction evidence beyond local prompting.

Direct correction is often unstable for ASR correction. In several settings, it produces higher CER than the original ASR hypotheses, suggesting that unconstrained LLM rewriting can introduce harmful edits. Our method mitigates this issue by grounding corrections in retrieved memory and structured dialogue evidence. For example, under zero-shot prompting, our method reduces C-CER from 36.40 to 31.76 with Qwen2.5-7B and from 35.66 to 29.04 with Qwen2.5-14B, corresponding to 12.7% and 18.6% relative reductions, respectively. For Qwen2.5-14B under zero-shot prompting, our method reduces C-CER from 35.66 to 29.04 and M-CER from 84.57 to 55.35, corresponding to relative reductions of 18.6% and 34.6%, respectively. Our method also improves cases where Direct is already competitive, such as the Qwen3.5-4B few-shot setting, where RelCER changes from -4.36% to 2.20%.

The Full-History baseline shows that simply exposing a long-context model to more historical text does not necessarily improve correction, as Gemma-4-26B-128K remains close to Raw ASR in both prompting settings. The gains are not uniform across all settings. Our method underperforms Direct in the Qwen3.5-9B few-shot setting, despite improving over Direct for the same backbone under zero-shot prompting. This indicates that few-shot demonstrations may interact with memory guidance and sometimes encourage unnecessary modifications under an inference-only setting. We therefore analyze correction behavior below to distinguish targeted improvements from harmful edits.

4.3 Analysis

Correction behavior analysis. CER measures final quality but hides how a model edits ASR hypotheses. We therefore use Edit Rate, Improve@Edit, and Worsen@Edit to measure whether edits occur, reduce CER, or increase CER. Figure 3 compares two representative backbones. Qwen2.5-14B illustrates a high over-editing case,

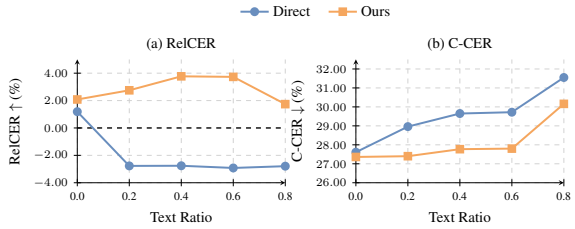

Figure 4: Impact of the text-input ratio on ASR correction performance.

Metric	Direct	Ours		
		Top-k=5	Top-k=10	Top-k=15
Subseq. Speech Window	4.59 s	4.59 s	4.59 s	4.59 s
User Wait Mean / P95	0.01 / 0.00 s	0.92 / 3.67 s	0.97 / 3.76 s	0.99 / 3.85 s
Visible Corr. Mean / P95	0.53 / 0.90 s	2.31 / 4.39 s	2.35 / 4.51 s	2.38 / 4.51 s
Backend Mean	0.53 s	2.98 s	3.04 s	3.08 s
RelCER ↑	1.19	1.37	2.08	2.05

Table 3: Online latency analysis under the Qwen3.5-4B zero-shot setting with continuous speech input.Ours is evaluated with different ontology retrieval sizes.

while Qwen3.5-4B represents a competitive setting where memory improves edit quality.

For Qwen2.5-14B, ontology memory mainly reduces excessive rewriting, although Worsen@Edit remains high after the model chooses to edit. For Qwen3.5-4B, ontology memory shows a more consistent pattern by reducing Edit Rate, increasing Improve@Edit, and decreasing Worsen@Edit in both prompting settings. These results suggest that ontology memory mainly serves as an edit-selection mechanism, while its effect on edit precision depends on how effectively the correction model uses the retrieved evidence. Appendix A.3 further analyzes how retrieval availability is associated with edit outcomes.

Impact of text-speech interleaving ratio. We study whether RAMC-Corr remains effective when subsequent utterances are provided in different modalities. Specifically, we vary the proportion of subsequent utterances represented as text transcriptions from 0.0 to 0.8, where 0.0 denotes the speech-only setting. Experiments are conducted with Qwen3.5-4B under the zero-shot setting. As shown in Figure 4, our method consistently achieves better RelCER and C-CER than Direct across all text ratios. The improvement becomes more pronounced after text transcriptions are introduced, especially at moderate ratios of 0.4 and 0.6. Direct, however, becomes less reliable under mixed text-speech inputs, leading to degraded RelCER. When the ratio increases to 0.8, the gain becomes smaller, indicating that more text context is not always more useful unless it provides correction-relevant cues. These results demonstrate the robustness of ontology memory under text-speech interleaved inputs.

Online latency in continuous speech interaction. We analyze latency in a continuous speech-input scenario, where backend computation for the previous chunk can overlap with the following speech segment and its ASR processing. Since this overlap can hide part of the computation from the user, we report user wait at the chunk level, together with backend latency and visible correction latency. Backend latency measures the end-to-end computation time for correction and memory operations, while visible correction latency measures the delay from ASR decoding to the corrected output. Direct denotes correction without working memory, while Ours denotes correction with memory retrieval and ontology update.

As shown in Table 3, Ours increases backend latency from 0.53 s to 2.98–3.08 s. However, the average user wait remains below 1 s across all Top-k settings because part of the computation is hidden by the 4.59 s speech-input window. Increasing Top-k from 5 to 15 only slightly increases average user wait from 0.92 s to 0.99 s, suggesting that larger retrieval sizes have limited impact on user-perceived latency in this setting.

Case study. Table 4 illustrates how ontology memory improves context-grounded ASR correction through retrieved evidence. Without such memory, the model tends to rely on linguistic plausibility, resulting in incomplete entity recovery or unsupported substitutions. For example, the Direcct fails to recover “院学生会” (graduate student union) from “院线汇” (cinema line association), and incorrectly rewrites the correct title “十宗罪” (The Ten Deadly Sins) as “失踪者” (The Missing). In contrast, our method grounds corrections in retrieved dialogue-relevant evidence, including organization context for “院学生会”, movie-title context for “悲伤逆流成河” (The Disillusioned Youth), and prior noise-form evidence for preserving “十宗罪”. These examples show that ontology memory supports more selective and contextually grounded edits rather than broader rewriting.

5 Related Work

Post-ASR correction improves transcription quality without modifying the underlying ASR system. Early methods relied on language model rescoring,

Num		Outputs	Memory Evidence
1	**Before**	真的你在院线汇不管你是院线汇还是西 *Really you are in the* ***cinema line association*** *whether you are in the* ***cinema line association*** *or west*	A retrieved memory about **Xiong Liang** (熊亮) mentions the graduate student union (院学生会), supporting the organization-name correction.
	After Direct	真的你在院线汇不管你是院线汇还是西 *Really you are in the* ***cinema line association*** *whether you are in the* ***cinema line association*** *or west*	
	After Ours	真的你在院学生会不管你是院学生会还是西 *Really you are in the* ***graduate student union*** *whether you are in the* ***graduate student union*** *or west*	
	Reference	真的你在院学生会不管你是在院学生会还是 *Really you are in the* ***graduate student union*** *whether you are in the* ***graduate student union*** *or*	
2	**Before**	就是这个悲伤有成河吗 *Is it this* ***sadness becomes a river***	A retrieved memory about **Yi Yao** (易遥) links the dialogue context to the movie title *The Disillusioned Youth* (悲伤逆流成河).
	After Direct	就是这个悲伤有长河吗 *Is it this* ***sadness has a long river***	
	After Ours	就是这个悲伤逆流成河吗 *Is it this* ***The Disillusioned Youth***	
	Reference	就是这个悲伤逆流成河吗 *Is it this* ***The Disillusioned Youth***	
3	**Before**	你看了是十宗罪呀 *What you watched was* ***The Ten Deadly Sins***	The retrieved memory for *The Ten Deadly Sins* (十宗罪) records prior ASR noise forms such as "missing crime" (失踪罪) and "Shi Zhongyu" (石中玉).
	After Direct	你看了是失踪者呀 *What you watched was* ***The Missing***	
	After Ours	你看了是十宗罪呀 *What you watched was* ***The Ten Deadly Sins***	
	Reference	你看的是十宗罪啊 *What you watched was* ***The Ten Deadly Sins***	

Table 4: Qualitative examples of ontology-memory-augmented ASR correction. **Before** shows the original ASR transcription. **After Direct** shows correction without ontology memory. **After Ours** shows correction with ontology memory. **Memory Evidence** shows the retrieved memory evidence provided as contextual support. Bold text highlights the key corrected or over-corrected expressions.

confusion modeling, and n-best hypothesis selection (Rubenstein et al., 2023; Rúnarsdóttir et al., 2019). With the development of large language models, recent studies have increasingly formulated ASR correction as a conditional generation task, where the model regenerates more accurate and fluent text conditioned on noisy ASR outputs (Yang et al., 2023; Chen et al., 2023). Despite their strong performance, most generative ASR correction methods still focus on the current utterance or local segment, with limited modeling of contextual knowledge accumulated from prior interactions.

To improve the correction of named entities, rare words, and domain-specific terminology, another line of research introduces contextual biasing or retrieval-augmented mechanisms that incorporate external vocabularies, entity databases, candidate phrases, or domain knowledge as auxiliary evidence for ASR correction (Zhang et al., 2023b; Ghosh et al., 2024). Ghosh et al. (2025) and Im et al. (2025) further combine semantic and phonetic similarity to alleviate generalization issues caused by open-vocabulary settings and acoustic confusion. These studies demonstrate the importance of external contextual evidence for ASR correction. However, most existing approaches still rely on static knowledge sources or independently retrieve candidate items for each utterance. In contrast, correction evidence in long-form interleaved text-speech conversations often naturally originates from prior interactions themselves.

As interaction duration increases, these limitations become more pronounced. Long-form conversational ASR must handle extended speech inputs, topic shifts, context truncation, cross-segment error propagation, and concept consistency. Recent studies have explored long-context speech understanding via benchmarks, speech compression, multimodal modeling, and long-context modeling (Yang et al., 2026; Ahn et al., 2026; AI, 2025; Wei et al., 2024). While these studies improve long-audio modeling, cross-segment continuity, and multimodal understanding, they pay relatively little attention to how textual and situational information established during prior interactions can be organized into reusable evidence for subsequent ASR correction.

Meanwhile, memory-augmented language models have been widely studied in tasks such as personalized dialogue, long-context question answering, and information compression (Ke et al., 2025; Chen et al., 2025). However, memory mechanisms have rarely been explored for post-ASR correction, especially in interleaved text-speech conversations where interaction context must be continuously accumulated, structurally organized, and dynamically retrieved for future correction.

6 Conclusion

We studied ASR correction in text-speech interleaved conversations, where long-range interaction history can provide valuable but noisy correction evidence. To exploit such context, we proposed an ontology-memory-augmented framework that organizes reliable dialogue history into a dynamically updatable working memory and retrieves structured evidence for subsequent ASR correction. We constructed RAMC-Corr from MagicData-RAMC to evaluate long-range context-grounded correction. Experiments across multiple LLM backbones show that our method improves over direct correction in 9 out of 10 paired settings. Further analyses indicate that these gains mainly come from more selective and evidence-grounded edits rather than indiscriminate rewriting, showing the potential of ontology memory for interactive ASR correction.

Limitations

Our work has four major limitations. First, RAMC-Corr is currently constructed from Chinese text-speech interleaved conversations. Although this setting provides a useful testbed for studying long-range context-grounded ASR correction, it does not fully cover multilingual, cross-lingual, or domain-diverse scenarios. In the future, we plan to extend the benchmark to more languages and real-world application domains.

Second, our experiments mainly focus on a training-free LLM setting with lightweight few-shot prompting. This design allows us to evaluate the plug-and-play capability of ontology memory across different LLM backbones. However, the potential benefits of supervised fine-tuning, instruction tuning, parameter-efficient adaptation, or more advanced prompting strategies remain under-explored. We leave a systematic investigation of these model adaptation strategies to future work.

Third, the retrieval component in our framework is not specifically optimized for ASR correction. The current retrieval strategy may not fully capture correction-specific cues such as phonetic similarity, entity recurrence, acoustic confusions, and discourse dependencies. Future work can develop task-aware retrievers that jointly consider semantic, phonetic, and conversational signals for more accurate context selection.

Finally, our evaluation is conducted on a benchmark derived from existing text-speech interleaved data. Although RAMC-Corr supports controlled evaluation of long-range context use, real-world interactive speech applications may involve more diverse speakers, background noise, spontaneous speech, and changing dialogue states. We plan to further validate the framework in more realistic and open-ended interaction scenarios.

Ethical Statement

We construct RAMC-Corr based on MagicData-RAMC for long-range context-grounded ASR correction in text-speech interleaved conversations. RAMC-Corr is derived from the existing MagicData-RAMC dataset and inherits its original speech data and reference transcripts. We do not collect new speech recordings, textual content, personal information, or private user data during dataset construction. The dataset is used only for research purposes, following the license and usage conditions of the original data source.

RAMC-Corr defines the evaluation split and correction targets for context-grounded ASR correction. The ASR hypotheses used in our experiments are generated by ASR systems during the experimental pipeline and are not newly collected human data. No personally identifiable information is introduced, and no additional sensitive or harmful content is added. The dataset is intended for evaluating context-grounded ASR correction rather than identifying or profiling individual speakers.

We acknowledge that automatic ASR correction may produce incorrect or unsupported modifications. Therefore, corrected transcripts should be used with caution in high-stakes scenarios, and human verification is recommended when they are used for consequential decisions.

Overall, the construction and use of RAMC-Corr are compliant with research ethics.

References

Hoseong Ahn, Jeongyun Chae, Yoonji Park, and Kyuhong Shim. 2026. Whisper-cd: Accurate long-form speech recognition using multi-negative contrastive decoding. *CoRR*, abs/2603.06193.

Mistral AI. 2025. Voxtral. *CoRR*, abs/2507.13264.

Blaine Billings and Bradley McDonnell. 2025. Connecting automated speech recognition to transcription practices. In *Proceedings of the Eight Workshop on the Use of Computational Methods in the Study of Endangered Languages*, pages 128–132, Honolulu, Hawaii, USA. Association for Computational Linguistics.

Chen Chen, Yuchen Hu, Chao-Han Huck Yang, Sabato Marco Siniscalchi, Pin-Yu Chen, and Chng Eng Siong. 2023. Hyporadise: An open baseline for generative speech recognition with large language models. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Nuo Chen, Hongguang Li, Jianhui Chang, Juhua Huang, Baoyuan Wang, and Jia Li. 2025. Compress to impress: Unleashing the potential of compressive memory in real-world long-term conversations. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 755–773, Abu Dhabi, UAE. Association for Computational Linguistics.

Sreyan Ghosh, Mohammad Sadegh Rasooli, Michael Levit, Peidong Wang, Jian Xue, Dinesh Manocha, and Jinyu Li. 2024. Failing forward: Improving generative error correction for ASR with synthetic data and retrieval augmentation. *CoRR*, abs/2410.13198.

Sreyan Ghosh, Mohammad Sadegh Rasooli, Michael Levit, Peidong Wang, Jian Xue, Dinesh Manocha, and Jinyu Li. 2025. Failing forward: Improving generative error correction for ASR with synthetic data and retrieval augmentation. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 2466–2482, Vienna, Austria. Association for Computational Linguistics.

HangChen HangChen, Chao-Han Huck Yang, Jia-Chen Gu, Sabato Marco Siniscalchi, and Jun Du. 2025. MISP-meeting: A real-world dataset with multimodal cues for long-form meeting transcription and summarization. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 15479–15492, Vienna, Austria. Association for Computational Linguistics.

Solee Im, Wonjun Lee, JinMyeong An, Yunsu Kim, Jungseul Ok, and Gary Lee. 2025. DeRAGEC: Denoising named entity candidates with synthetic rationale for ASR error correction. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 15181–15193, Vienna, Austria. Association for Computational Linguistics.

Cai Ke, Yiming Du, Bin Liang, Yifan Xiang, Lin Gui, Zhongyang Li, Baojun Wang, Yue Yu, Hui Wang, Kam-Fai Wong, and Ruifeng Xu. 2025. Flexibly utilize memory for long-term conversation via a fragment-then-compose framework. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 21119–21136, Suzhou, China. Association for Computational Linguistics.

Heeseung Kim, Che Hyun Lee, Sangkwon Park, Jiheum Yeom, Nohil Park, Sangwon Yu, and Sungroh Yoon. 2025. Does your voice assistant remember? analyzing conversational context recall and utilization in voice interaction models. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 8984–9014, Vienna, Austria. Association for Computational Linguistics.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. In *Proceedings of the 29th Symposium on Operating Systems Principles, SOSP 2023, Koblenz, Germany, October 23-26, 2023*, pages 611–626. ACM.

Paul K. Rubenstein, Chulayuth Asawaroengchai, Duc Dung Nguyen, Ankur Bapna, Zalán Borsos, Félix de Chaumont Quitry, Peter Chen, Dalia El Badawy, Wei Han, Eugene Kharitonov, Hannah Muckenhirn, Dirk Padfield, James Qin, Danny Rozenberg, Tara N. Sainath, Johan Schalkwyk, Matthew Sharifi, Michelle Tadmor Ramanovich, Marco Tagliasacchi, Alexandru Tudor, Mihajlo Velimirovic, Damien Vincent, Jiahui Yu, Yongqiang Wang, Vicky Zayats, Neil Zeghidour, Yu Zhang, Zhishuai Zhang, Lukas Zilka, and Christian Havnø Frank. 2023. Audiopalm: A large language model that can speak and listen. *CoRR*, abs/2306.12925.

Anna V. Rúnarsdóttir, Inga Rún Helgadóttir, and Jón Guðnason. 2019. Lattice re-scoring during manual editing for automatic error correction of ASR transcripts. In *20th Annual Conference of the International Speech Communication Association, Interspeech 2019, Graz, Austria, September 15-19, 2019*, pages 3810–3814. ISCA.

Kun Wei, Bei Li, Hang Lv, Quan Lu, Ning Jiang, and Lei Xie. 2024. Conversational speech recognition by learning audio-textual cross-modal contextual representation. *IEEE ACM Trans. Audio Speech Lang. Process.*, 32:2432–2444.

Chao-Han Huck Yang, Yile Gu, Yi-Chieh Liu, Shalini Ghosh, Ivan Bulyko, and Andreas Stolcke. 2023. Generative speech recognition error correction with large language models and task-activating prompting. In *IEEE Automatic Speech Recognition and Understanding Workshop, ASRU 2023, Taipei, Taiwan, December 16-20, 2023*, pages 1–8. IEEE.

Fei Yang, Xuanfan Ni, Renyi Yang, Jiahui Geng, Qing Li, Chenyang Lyu, Yichao Du, Longyue Wang, Weihua Luo, and Kaifu Zhang. 2026. Longspeech: A scalable benchmark for transcription, translation and understanding in long speech. *CoRR*, abs/2601.13539.

Zehui Yang, Yifan Chen, Lei Luo, Runyan Yang, Lingxuan Ye, Gaofeng Cheng, Ji Xu, Yaohui Jin, Qingqing Zhang, Pengyuan Zhang, Lei Xie, and Yonghong Yan. 2022. Open source magicdata-ramc: A rich annotated mandarin conversational(ramc) speech dataset. In *23rd Annual Conference of the International Speech Communication Association, Interspeech 2022, Incheon, Korea, September 18-22, 2022*, pages 1736–1740. ISCA.

Dong Zhang, Shimin Li, Xin Zhang, Jun Zhan, Pengyu Wang, Yaqian Zhou, and Xipeng Qiu. 2023a. SpeechGPT: Empowering large language models with intrinsic cross-modal conversational abilities. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 15757–15773, Singapore. Association for Computational Linguistics.

Min Zhang, Xiaosong Qiao, Yanqing Zhao, Chang Su, Yinglu Li, Yuang Li, Ming Zhu, Mengyao Piao, Song Peng, Shimin Tao, Hao Yang, and Yanfei Jiang. 2023b. Knowledge prompt for whisper: An ASR entity correction approach with knowledge base. In *IEEE International Conference on Big Data, BigData 2023, Sorrento, Italy, December 15-18, 2023*, pages 2975–2979. IEEE.

A Appendix

A.1 RAMC-Corr Dataset Construction Details

This section provides additional details on the sample construction process of RAMC-Corr. Given an

original conversation from MagicData-RAMC, we select a split boundary from its temporally ordered speech segments and manual transcriptions. The manual transcriptions before the boundary are used as the trusted prefix context, while the speech segments after the boundary form the target region to be corrected. The boundary selection process is based only on manual transcription texts, ensuring that the split does not depend on ASR outputs from the target region.

For a conversation containing n segments, we denote its manual transcription sequence as:

$$Y = \{y_i\}_{i=1}^{n},$$

where y_i denotes the manual reference transcription of the i-th segment. A candidate boundary b divides the sequence into:

$$G_b = \{y_i\}_{i=1}^{b-1},$$

$$R_b = \{y_i\}_{i=b}^{n}.$$

Here, G_b denotes the trusted prefix context before the boundary, and R_b denotes the reference transcriptions corresponding to the target region after the boundary. The trusted prefix always starts from the beginning of the conversation and continuously extends to the segment immediately before the boundary. For each valid conversation, we retain only one final boundary, so that each conversation produces at most one RAMC-Corr sample. This avoids generating multiple overlapping samples from the same conversation.

A.1.1 Candidate Boundary Generation

We first enumerate candidate boundaries over the segment sequence of each conversation and apply preliminary filtering based on length and position constraints. Specifically, the trusted prefix before the boundary is required to contain a sufficient number of valid semantic segments and valid characters, so that it can provide usable historical context. At the same time, the trusted prefix must not exceed a predefined upper bound, in order to control the input size and reduce redundant history. The target region after the boundary is also required to satisfy minimum constraints on the number of segments and text length, ensuring that each sample contains a multi-segment correction target.

In addition, we constrain the relative position of the candidate boundary within the full conversation. This prevents the boundary from being placed too early, which would result in insufficient context, or too late, which would make the target region too short. Candidate boundaries that satisfy these constraints are then passed to the context continuity evaluation stage.

A.1.2 Text Validity Filtering

When computing length statistics and similarity scores, we only consider text segments with clear semantic content. Specifically, we filter out empty texts, noise markers, symbol-only segments, segments with an excessively high punctuation ratio, and short backchannel expressions commonly observed in real conversations.

This filtering is used only for statistics and similarity computation during boundary selection. It does not modify the original segment sequence in the final sample. The final samples still preserve the original conversation order, together with the corresponding speaker and timestamp information.

A.1.3 Context Continuity Constraint

To ensure that the trusted prefix and the target region remain semantically coherent, we compute a context continuity score for each candidate boundary. Specifically, we concatenate the valid texts in G_b as the prefix text, and concatenate the valid texts from the first K segments after the boundary as the target-head text H_b. We then compute the similarity between these two texts.

We use a weighted fusion of character n-gram similarity and semantic embedding similarity:

$$s_b = \alpha s_{\text{ngram}}(G_b, H_b) + (1-\alpha) s_{\text{emb}}(G_b, H_b),$$

where s_{ngram} measures entity, keyword, and local lexical overlap, while s_{emb} measures topic-level semantic relatedness. The coefficient α is the fusion weight. We retain only candidate boundaries whose s_b reaches the predefined threshold, and use this score as an important criterion for final boundary selection.

A.1.4 Final Boundary Selection

After filtering by length, position, and context continuity constraints, each conversation may retain multiple candidate boundaries. We further select one final boundary from these candidates to construct the corresponding RAMC-Corr sample. The selection process prioritizes context continuity across the boundary, while also preferring a trusted prefix with appropriate length and a reasonable

Parameter	Value	Description
embedding model	Qwen3-Embedding-8B	Embedding model
min prefix segments	5	Min. valid prefix segments
min prefix chars	80	Min. valid prefix chars
max prefix segments	50	Max. valid prefix segments
max prefix chars	1200	Max. valid prefix chars
min target segments	10	Min. valid target segments
min target chars	150	Min. valid target chars
target head K	8	Target-head segments
min similarity	0.25	Min. continuity score
min boundary ratio	0.05	Min. boundary position
max boundary ratio	0.30	Max. boundary position
target prefix segments	20	Expected prefix length
min boundary chars	4	Min. chars at boundary start
fusion weight α	0.4	n-gram fusion weight
w_s	0.70	Continuity weight
w_l	0.20	Length penalty weight
w_r	0.10	Position penalty weight

Table 5: Parameters used in the construction of RAMC-Corr. Min. and Max. denote minimum and maximum, respectively; chars denotes characters.

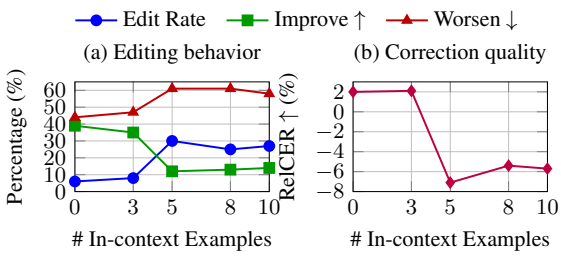

Figure 5: Effect of the number of in-context examples on editing behavior and correction quality.

Setting	Retrieval	Count	Edit	Improve ↑	Worsen ↓
Zero-shot	With	13,304	6.16	39.19	42.98
Zero-shot	Without	402	9.20	35.14	54.05
Few-shot	With	13,106	7.55	35.89	45.60
Few-shot	Without	600	10.17	26.23	59.02

Table 6: Edit outcomes grouped by retrieval availability for Qwen3.5-4B. Edit denotes Edit Rate, while Improve and Worsen denote Improve@Edit and Worsen@Edit. All rates are percentages.

boundary position. This ensures that each sample contains sufficient historical context while preserving enough subsequent content for correction.

For the candidate boundary b of the c-th conversation, we define its selection score as:

$$S_c(b) = w_s s_c(b) - w_l d_c(b) - w_r r_c(b).$$

Here, $s_c(b)$ denotes the context continuity score across the boundary; $d_c(b)$ denotes the deviation between the trusted prefix length and the expected prefix length; $r_c(b)$ denotes the boundary position penalty, which is used to avoid boundaries that are too early or too late; and w_s, w_l, and w_r are the corresponding weights. This scoring function favors split points with stronger context continuity, a moderately sized trusted prefix, and a reasonable boundary position.

The final boundary is defined as:

$$b_c^* = \arg\max_{b \in \mathcal{B}_c} S_c(b),$$

where $\mathcal{B}_c$ denotes the set of candidate boundaries for the c-th conversation after length, position, and context continuity filtering. For each valid conversation, we retain only the boundary with the highest score. Therefore, each conversation produces at most one RAMC-Corr sample. If no candidate boundary satisfies the constraints, the conversation is excluded from the construction of RAMC-Corr. The specific values of each parameter are shown in the table 5.

A.2 Effect of in-context examples.

We further study the effect of adding in-context examples to the correction prompt. To construct the example pool, we first sampled 10 conversations from the training set and then sampled 200 utterances from these conversations. From this pool, we manually selected and annotated 10 representative ASR errors as candidate in-context examples.

As shown in Figure 5, simply increasing the number of examples does not consistently improve correction quality. While using a small number of examples yields positive relative CER reduction, adding more examples makes the model edit more aggressively. In particular, when the number of examples increases from 3 to 5, the edit rate rises substantially, but the proportion of worsened edits also increases, leading to a sharp drop in relative CER reduction. This suggests that naively adding more static examples can induce over-correction, where the model edits more often but introduces more harmful changes.

These results indicate that the construction of in-context examples remains an important design factor. More effective example selection, retrieval, and ordering strategies may be needed to ensure that demonstrations provide useful correction guidance rather than biasing the model toward unnecessary edits.

A.3 Evidence Utilization Analysis

We compute all statistics at the correction-record level. Retrieval availability indicates whether at least one ontology entry is retrieved and provided to the correction model. A record is counted as edited if the corrected output differs from the ASR hypothesis. Improve@Edit and Worsen@Edit are

computed among edited records, measuring the proportions of edits that decrease or increase CER, respectively. Memory updates are detected when the ontology working memory is modified after processing the current record.

Table 6 analyzes edit outcomes by retrieval availability for Qwen3.5-4B. Retrieved ontology evidence is available for 97.07% and 95.62% of records under zero-shot and few-shot prompting, respectively, with memory updates detected in 6.48% and 6.68% of records. Across both settings, records with retrieved evidence show lower Edit Rate and Worsen@Edit than those without retrieval, suggesting more conservative and less harmful editing.

B Prompt Templates

This section provides the prompt templates used in our experiments. Table 7 shows the correction prompt without ontology working memory, Table 8 shows the memory-augmented correction prompt, and Table 9 shows the prompt for extracting candidate ontology entries.

ASR Correction without Ontology Memory

User: You are an ASR text correction expert. Your task is to identify words in the current transcription that need correction based on the current ASR text and recent dialogue context, and output the corrected text.

Task priorities:
1. Prioritize ASR correction and make the corrected text as accurate as possible.
2. Only correct ASR recognition errors. Do not polish, summarize, expand, or rewrite the style.
3. If there is no clear error, keep the corrected text identical to the current transcription.

Correction principles:
- First inspect the current sentence, then refer to the historical context.
- Focus on keywords, named entities, terms, numbers, collocations, and contextually inconsistent words.
- Do not change the topic based on a single abrupt word unless the context clearly indicates a topic shift.
- If the context does not provide enough evidence, keep the original word.
- Preserve the original word order, spoken style, and amount of information as much as possible.

Forbidden behaviors:
- Do not turn correction into polishing, summarization, or rewriting.
- Do not add information that is not expressed in the current sentence.
- Do not delete semantically meaningful spoken components.
- Do not output explanatory completion text.

Recent dialogue context:
{recent_corrected_context}

Current ASR transcription:
{current_asr_text}

Please output only the corrected text.

LLM: {corrected_text}

Table 7: Prompt template for ASR correction without ontology memory.

ASR Correction with Ontology Memory

User: You are an ASR text correction expert. Correct the current transcription using recent dialogue context, active entities, and retrieved ontology working memory (WM).

Task priorities:
...

Correction principles:
...
- Inspect the current sentence first, then recent context, and finally WM.
- If a word conflicts with recent topic, active entities, or WM, correct it only after confirming it is ASR-suspicious.
- If neither context nor WM provides enough evidence, keep the original word.
- Recent contextual evidence has priority over distant WM.

WM usage rules:
- WM is auxiliary evidence, not a forced replacement basis.
- Retrieved knowledge may contain noise.
- Do not replace text with a WM entity solely because it appears in WM.
- Use WM only when the ASR text is suspicious or contextually inconsistent.
- If WM conflicts with the current sentence and no clear ASR error exists, trust the current sentence.

Forbidden behaviors:
...
- Do not force the sentence to match WM or change the topic for WM.
- Do not change ordinary words into WM entities or terms without evidence.

Inputs:
Retrieved WM: {retrieved_knowledge}
Active entities: {active_mentions}
Recent context: {recent_corrected_context}
Current ASR: {current_asr_text}

LLM: {corrected_text}

Table 8: Prompt template for ASR post-correction with retrieved ontology memory and recent active entities. Shared instructions inherited from the no-memory prompt are omitted for brevity, while WM-specific constraints are shown.

Ontology Instruction

Role Definition
The model acts as a working-memory update expert for an ASR correction system. Its task is to extract stable nodes that can be safely written into working memory (WM) based on the original ASR transcript `transcript_text` and the corrected text `corrected_text`. These nodes should be clearly useful for subsequent ASR correction. The output must be strictly valid JSON.

Core Objective
This task is not intended to build a complete knowledge graph, nor to summarize the text content. It should only record anchors that explicitly appear in `corrected_text`, are stable and reusable, and are clearly helpful for future ASR correction.

A valid node must satisfy:
1. The `canonical_name` must come from a phrase explicitly appearing in `corrected_text`.
2. It must be a stable and reusable entity, term, special identifier, special version, code, abbreviation, proprietary expression, work title, organization name, role name, group name, product name, domain term, core concept, or key numerical value.
3. If it is misrecognized, incorrectly segmented, omitted, inserted, or mistranscribed by ASR in the future, it should significantly affect the semantics or character error rate (CER).
4. If there is a corresponding erroneous fragment in `transcript_text`, it should be recorded as an `asr_noise_form`.
5. It must not be an ordinary word, ordinary phrase, sentence-level rewrite, semantic summary, polishing completion, or temporary description.

Recent context: {history/context}
Original ASR transcription: {transcript_text}
Corrected text: {corrected_text}
Output format:
Please output only valid JSON with the field `ontology_nodes`. If no valid node exists, output an empty list.

Table 9: Prompt template for ontology extraction.