Cognitive Scaffold: From Fluid Context to Crystallized Memory for Long-Horizon DeepResearch Agents

Qiuyuan Ai[1]*, Zenghuang Fu[2]*, Zhaoyang Li[1]*, Ping Jiang[1,3], Haoyu Wu[1,3], Jie Song[1]†, Guannan He[1]†
[1]Peking University
[2]Institute of Automation, Chinese Academy of Sciences
[3]Mininglamp Technology
gnhe@pku.edu.cn, jie.song@pku.edu.cn

Abstract

Scaling LLM-based agents to long-horizon deep research is constrained by the "context-noise trade-off," where linear history accumulation degrades reasoning and dilutes fine-grained evidence. To address this, we introduce the **Cognitive Scaffold**, a factorized memory architecture that decouples the cognitive state into a Fluid Working Context for immediate reasoning and a persistent Knowledge Graph for long-term retention. Unlike unstructured summarization, our framework employs a Rejection Sampling Fine-Tuning (RFT) pipeline to crystallize saturated context into structured "event snapshots," strictly enforcing atomic constraints to preserve numerical values and entities. During reasoning, a thought-driven dual-path retrieval mechanism enables the agent to proactively recover precise evidence. Empirical evaluations on Xbench-DeepSearch, BrowseComp-ZH, and GAIA demonstrate that Cognitive Scaffold consistently outperforms baselines, achieving 74.7% Avg@3 and 87.0% Pass@3 on Xbench-DeepSearch, 48.5% Avg@3 and 65.9% Pass@3 on BrowseComp-ZH, and 72.8% Avg@3 and 88.3% Pass@3 on GAIA, while reducing compression hallucinations to 5.3%. We open-source our codebase to facilitate future research.

1 Introduction

Modern scientific and industrial progress increasingly depends on the ability to seek, verify, and synthesize information from the web and heterogeneous corpora. Recent LLM-based agents have therefore shifted the paradigm from passive retrieval-augmented generation (RAG) to autonomous, tool-augmented research that iteratively plans, searches, executes, and consolidates evidence before producing a final report (Schick et al., 2023; Wang et al., 2023; Nan et al., 2025). Such deep research workloads are inherently long-horizon: solving a single query may require hundreds of interactions with noisy environments (e.g., raw HTML, search results, code repositories), and the agent must retain and revisit precise evidence across extended trajectories.

However, scaling LLM agents to longer horizons is fundamentally constrained by the prevailing design of agentic frameworks: they linearly accumulate the entire interaction history (reasoning traces, tool calls, and observations) into a single, ever-growing context buffer. This induces a severe context-noise trade-off (Yen et al., 2025; Liu et al., 2024). As the history length T grows far beyond the context limit $\mathcal{L}_{\max}$, agents suffer from (i) cognitive degradation, where relevant evidence becomes "lost in the middle" and decisions increasingly rely on parametric priors rather than retrieved facts (Liu et al., 2024); and (ii) computational inefficiency, since attention and KV-cache management costs grow rapidly with context length (Press et al., 2021). In practice, these effects manifest as brittle planning, evidence being ignored, and unstable tool-use behaviors under dense web noise.

A natural direction is to compress history. Summary-based methods periodically replace the overflowing context with a short natural-language synopsis (Wu et al., 2025b), while more recent folding-style approaches attempt to proactively condense completed sub-trajectories (Ye et al., 2025; Yan et al., 2025; Fang et al., 2025). Although effective at reducing token count, these approaches remain largely linear and irreversible: once dense evidence is compressed, fine-grained details—notably numerical values and named entities that are critical for scientific reasoning—are easily lost (Durmus et al., 2020). Multi-agent decompositions offer another route by distributing context across specialized agents (Zhang et al., 2025b), but they typically rely on handcrafted, task-specific workflows and resist end-to-end optimiza-

*Equal contribution.
†Corresponding author.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 25526–25542

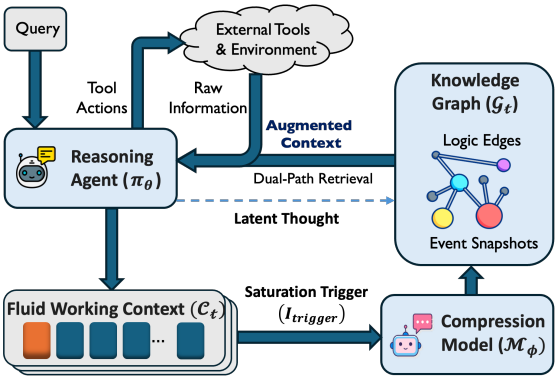

Figure 1: **The Cognitive Scaffold Architecture.** The framework decouples memory into a fluid Working Context ($\mathcal{C}_t$) and a persistent Knowledge Graph ($\mathcal{G}_t$). Upon saturation ($\mathbb{I}_{trigger}$), the Compression Model ($\mathcal{M}_\phi$) crystallizes history into event snapshots. During reasoning, the agent (π_θ) employs latent thoughts to trigger dual-path retrieval, ensuring long-horizon consistency.

tion. Consequently, current systems still lack a principled mechanism that simultaneously (a) bounds the active context, (b) preserves high-fidelity evidence, and (c) supports structured revisitation of past decisions.

In this work, we propose the Cognitive Scaffold (Figure 1), a factorized memory mechanism designed to transcend these limitations. As illustrated in the figure, unlike existing agents that rely solely on a transient feedback loop (top-left), our architecture introduces a persistent memory cycle (bottom-right). The key idea is to decouple the agent state into a Fluid Working Context ($\mathcal{C}_t$) for immediate reasoning and a persistent Knowledge Graph ($\mathcal{G}_t$) for long-term retention. Instead of treating memory as a monolithic token sequence, our agent periodically crystallizes saturated segments of $\mathcal{C}_t$ into graph nodes that store high-fidelity Event Snapshots (s_t), using a specialized compression model ($\mathcal{M}_\phi$). When reasoning gaps are detected, the reasoning backbone (π_θ) proactively recovers relevant sub-graphs via introspective, thought-driven navigation. To ensure factual precision during consolidation, we introduce a Rejection Sampling Fine-Tuning (RFT) pipeline that enforces atomic constraints over numerical values and entities, substantially reducing hallucinations. Finally, we utilize a specialized lightweight compressor and a Knowledge Graph to ensure efficient reasoning within a bounded context.

We empirically evaluate our approach on three long-horizon benchmarks spanning general reasoning and bilingual web navigation. As shown in Table 1, the proposed scaffold consistently outperforms the LLM-based React Agent and the conventional DeepResearch Agent, while improving efficiency by bounding the active context and delegating long-term retention to the graph.

Our contributions are threefold:

- We introduce Cognitive Scaffold, a factorized memory design that decouples short-term reasoning from long-term retention via a dynamic event-centric Knowledge Graph.
- We propose an RFT-based consolidation mechanism with an atomic fidelity constraint, enabling high-precision compression that preserves numerical values and entities. Furthermore, we introduce Automated Prompt Optimization (APO), a compressor-in-the-loop procedure that selects a model-specific compression guideline for each distilled compressor, improving instruction following and stability at deployment.
- We implement a Thought-Driven Dual-Path Retrieval strategy and a decoupled inference system. By performing retrieval via latent thoughts and merging context via inter-turn injection, our framework supports scalable long-horizon deep research, achieving strong performance across diverse benchmarks.

2 Related Work

2.1 Deep Research and Autonomous Information Seeking

The landscape of information retrieval is shifting from passive Retrieval-Augmented Generation (RAG) to autonomous Deep Research Agents. Unlike traditional RAG, which performs "one-shot" retrieval, Deep Research Agents emulate human workflows through iterative planning, execution, and reflection. Proprietary systems such as OpenAI Deep Research (OpenAI, 2025a) and Gemini Deep Research (Team, 2025a) already exhibit strong performance on open-ended inquiries, often executing long tool-use trajectories before producing a final report.

In the open-source community, WebResearcher (Qiao et al., 2025) formulates deep research as a Markov Decision Process (MDP), while InfoAgent (Zhang et al., 2025a) and WebDancer (Wu et al., 2025a) enhance autonomy via self-refining search strategies, typically within a "Plan-Act-Reflect"

loop (Yao et al., 2022). However, these systems largely treat interaction history as a flat text buffer: reasoning traces, search logs, and notes are linearly appended into the context window. As horizons extend to hundreds of steps (e.g., in BrowseComp-style benchmarks), this design leads to Context Suffocation: looking back becomes costly and it is hard to isolate or revise specific branches of the research process. Our work instead views the trajectory as a structured scaffold of research events rather than a monolithic token sequence.

2.2 Dynamic Context Management in Long-Horizon Tasks

To mitigate context saturation, recent work has moved beyond static windows toward dynamic context management. ReSum (Wu et al., 2025b) introduces a "reset-and-summarize" mechanism, while Context Folding (Sun et al., 2025) utilizes branching and folding to manage tasks. AgentFold (Ye et al., 2025) adds a proactive "folding" action to selectively condense sub-trajectories. Distinctively, HiAgent (Hu et al., 2025) mimics human memory by segmenting history into subgoal-based chunks, maintaining detailed context only for the active subgoal while compressing completed ones into textual observations.

However, a fundamental limitation persists across these approaches: they rely on textual abstraction as the primary storage medium. Whether through periodic summarization (ReSum) or subgoal chunking (HiAgent), converting rich interaction logs into natural language summaries inevitably introduces semantic blurring (Durmus et al., 2020). Critical details—such as specific error codes, numerical parameters, or intermediate entity states—are often smoothed out in the text generation process. Furthermore, retrieving information from a sequence of textual summaries still requires linear scanning, which struggles with multi-hop reasoning. In contrast, our Cognitive Scaffold factorizes the cognitive state. Instead of piling up text summaries, we crystallize evidence into a structured Knowledge Graph, enabling precise, non-linear access to atomic facts without the resolution loss inherent in textual compression.

2.3 Structured Cognitive Memory and Scaffolding

Our work draws on Neuro-Symbolic Architectures (Chandra et al., 2025; Gershman et al., 2025) that separate stable structural scaffolds from high-fidelity content. In the LLM domain, systems like GraphRAG (Edge et al., 2024) and Graphiti (Rasmussen et al., 2025) leverage Knowledge Graphs for retrieval. While recent iterations support incremental updates, they primarily function as Declarative Memory—indexing external corpora to map pre-existing world knowledge. Crucially, they remain passive query engines lacking the operational protocols for autonomous Deep Research, such as determining what to crystallize and when to trigger retrieval endogenously.

In contrast, we propose the Cognitive Scaffold as a dynamic mechanism for Knowledge Accumulation. Unlike corpus-centric indexing, our knowledge graph evolves in real-time to crystallize synthesized insights and web summaries derived from the agent's exploration. Instead of raw interaction logs, it stores compressed high-value evidence (e.g., key conclusions from visited pages), effectively functioning as a dynamic research notebook that grows with the inquiry. This design allows agents to maintain a global view of collected evidence while retaining local precision, effectively functioning as a digital hippocampus for infinite-horizon reasoning.

3 Methodology

We introduce the **Cognitive Scaffold**, a neuro-symbolic architecture that mitigates linear context limits by decoupling the agent's state into a *Fluid Working Context* ($\mathcal{C}_t$) for reasoning and a persistent *Knowledge Graph* ($\mathcal{G}_t$) for retention. This factorization prevents context saturation (Figure 2). We formalize the framework in §3.1, detail the Rejection Sampling Fine-Tuning (RFT) pipeline for high-fidelity compression in §3.2, and present the Introspective Retrieval mechanism in §3.3.

3.1 The Cognitive Scaffold Framework

Formally, we instantiate the Deep Research Agent as a dynamic system evolving over discrete interaction steps t. The system is driven by two specialized models: a primary Reasoning Agent (π_θ) for high-level planning and a Compression Model ($\mathcal{M}_\phi$) for context consolidation. As illustrated in Figure 2, the framework operates through a continuous evolutionary process.

Stage 1: Fluid Reasoning ($\mathcal{C}_t$). The process begins in the Fluid Phase, where π_θ performs linear reasoning within the Fluid Working Context $\mathcal{C}_t$. This memory acts as a sliding window for recent tokens.

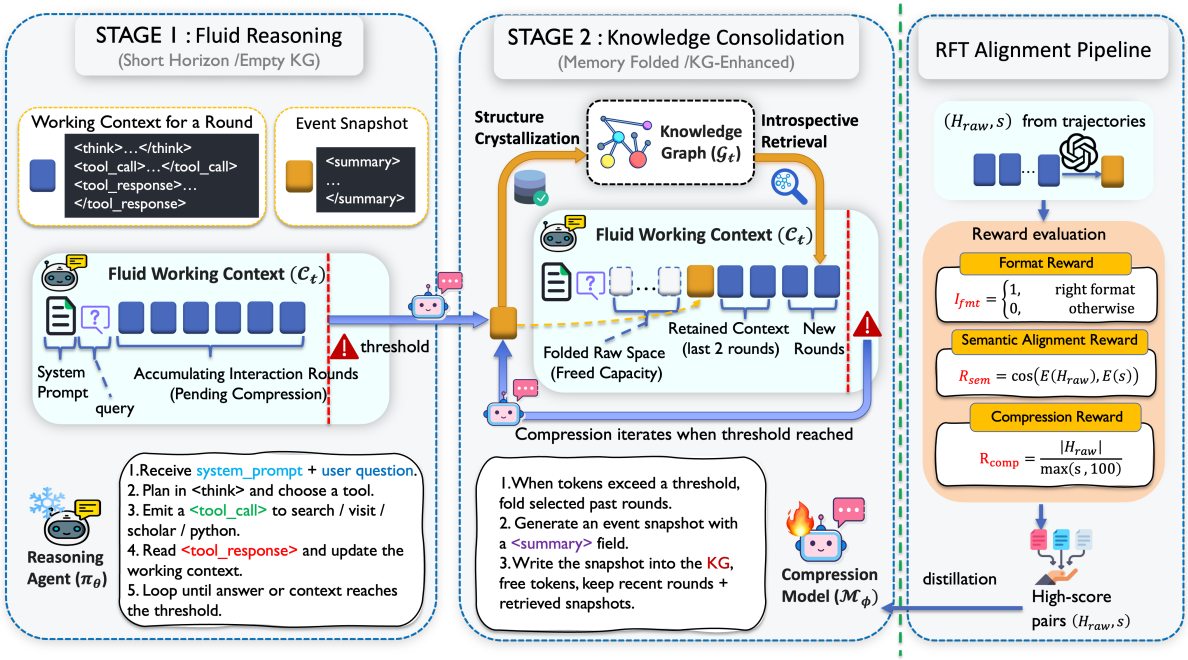

Figure 2: The evolutionary process of the Cognitive Scaffold. The framework operates in two recurrent stages. Left (Fluid Reasoning): The Reasoning Agent (π_θ) interacts with the environment within a transient Fluid Working Context ($\mathcal{C}_t$). Transition: When the context fills, the Compression Model ($\mathcal{M}_\phi$) crystallizes overflowed history into structured Event Snapshots. Middle (Knowledge Consolidation): These snapshots are stored in the persistent Knowledge Graph ($\mathcal{G}_t$), enabling Introspective Retrieval where the agent actively queries past information to support current reasoning via the structure crystallization and retrieval loop. Right (RFT Alignment): The compression model is trained via a Rejection Sampling Fine-Tuning pipeline that enforces atomic constraints through format, semantic, and compression rewards.

Let Q denote the initial user query and Δ_t represent the interaction increment at step t (containing the thought chain, tool execution, and observation). In this phase, the context accumulates linearly:

$$\mathcal{C}_{t+1} \leftarrow \mathcal{C}_t \oplus \Delta_t \tag{1}$$

As interactions accumulate and $|\mathcal{C}_t|$ approaches its capacity limit $\tau(k)$ (where k denotes the number of prior folding events), the system triggers a state transition.

Structure Crystallization (Update & Lazy Injection). Upon saturation ($\mathbb{I}_{trigger} = 1$), the overflowed history segment is transferred to $\mathcal{M}_\phi$ for the Crystallization Process. Using the RFT pipeline, $\mathcal{M}_\phi$ extracts the essential information from the noisy stream and converts it into the k-th Event Snapshot (s_k). This snapshot serves a dual purpose via a unified Lazy Injection protocol: (i) Storage: s_k is structured into a node v_k and permanently committed to the Knowledge Graph: $\mathcal{G}_{t+1} \leftarrow \mathcal{G}_t \cup \{v_k\}$. (ii) Anchor Injection: To ensure context continuity without retaining the full history, s_k is injected back into the cleared working memory as a structured "Memory Note".

Instead of silently deleting history, the updated context for the next step $t+1$ is reconstructed as:

$$\mathcal{C}_{t+1} \leftarrow \mathcal{C}_{\text{head}} \oplus \text{NOTE}(s_k) \oplus \mathcal{H}_{\text{recent}} \tag{2}$$

Here, $\mathcal{C}_{\text{head}}$ contains the immutable system prompt and the original user query Q. $\mathcal{H}_{\text{recent}}$ is a buffer of the most recent interaction rounds retained to preserve local coherence. This ensures the agent explicitly perceives the consolidated insight alongside the original goal, effectively bridging the past and future.

Stage 2: Knowledge-Enhanced Reasoning ($\mathcal{G}_t$). Following the transition, the system enters the Structured Phase. The Reasoning Agent π_θ resumes with the refreshed window, but now possesses the capability of Introspective Retrieval. The Knowledge Graph $\mathcal{G}_t$ serves as an external memory bank. If a retrieval operation yields a set of nodes and edges (packaged as OBS_{ret}), the context update rule is augmented:

$$\mathcal{C}_{t+1} \leftarrow \mathcal{C}_t \oplus \Delta_t \oplus \text{OBS}_{\text{ret}} \tag{3}$$

This completes the system's dynamic cycle: (1) *accumulate* linearly (Eq. 1), (2) *crystallize* upon

saturation (Eq. 2), and (3) *augment* via retrieval (Eq. 3). The full inference procedure is detailed in Algorithm 1 in Appendix B.

3.2 High-Fidelity Memory Compression via RFT

The effectiveness of the Cognitive Scaffold hinges on the quality of the crystallized nodes. The Compression Model $\mathcal{M}_\phi$ must function not merely as a summarizer, but as a high-fidelity information distiller. To achieve this, we employ a Rejection Sampling Fine-Tuning (RFT) pipeline alongside a dynamic triggering mechanism.

Dynamic Density Triggering. Instead of fixed-interval compression, the crystallization process is governed by the information density of the Fluid Working Context $\mathcal{C}_t$. We monitor the token accumulation and define the trigger condition as $\mathbb{I}_{\text{trigger}} = \mathbb{I}\left(|\mathcal{C}_t| > \tau(k)\right)$, where k is the count of prior folding events. To prevent premature compression of complex dependency chains in later stages, the threshold $\tau(k)$ adaptively expands following a backoff logic:

$$\tau(k) = \min\left(\tau_{base} + \left\lfloor \frac{k}{\eta} \right\rfloor \cdot \Delta\tau, \mathcal{L}_{max}\right) \quad (4)$$

Here, τ_{base} is the initial capacity, η controls the expansion frequency, and $\mathcal{L}_{max}$ is the context ceiling. This ensures the active buffer grows to accommodate longer reasoning traces as the task complexity increases.

The RFT Alignment Pipeline. To train $\mathcal{M}_\phi$, we construct a dataset of (H_{raw}, s) pairs, where H_{raw} is a raw history segment and s is a high-quality event snapshot. We adopt a "Generate-Evaluate-Distill" protocol: a teacher model generates candidate snapshots, which are then filtered by a composite reward function. Unlike strictly rule-based filtering, we employ a weighted scoring metric to balance structural correctness, semantic fidelity, and compression rate:

$$\begin{aligned} Score(s) = {} & w_{fmt} \cdot \mathbb{I}_{fmt}(s) \\ & + w_{sem} \cdot R_{sem}(H_{raw}, s) \\ & + w_{comp} \cdot R_{comp}(H_{raw}, s) \end{aligned} \quad (5)$$

The components are defined as follows: (i) Format Consistency ($\mathbb{I}_{fmt}$): A binary indicator checking strict adherence to the output format (e.g., proper XML tags). (ii) Semantic Alignment (R_{sem}): We measure information preservation via the cosine similarity between embeddings: $R_{sem} = \cos(E(H_{raw}), E(s))$. (iii) Compression Efficiency (R_{comp}): We incentivize conciseness using the reduction ratio, penalized by a length floor: $R_{comp} = |H_{raw}| / \max(|s|, 100)$. Based on our empirical tuning (as shown in the accompanying code), we set the weights to $w_{fmt} = 0.1$, $w_{sem} = 10$, and $w_{comp} = 0.02$, prioritizing semantic fidelity above all else. High-scoring pairs are retained to form the distillation dataset $\mathcal{D}_{train}$.

Automated Prompt Optimization (APO). Since the optimal compression guideline is often model-dependent, we further refine the prompt for $\mathcal{M}_\phi$ using a black-box search. We treat Eq. (5) as the objective function and iteratively propose guideline variants, evaluating them on held-out trajectories to select the instruction that maximizes the composite reward.

Graph Instantiation via Graphiti. The output of the compression pipeline is a dense natural language summary (the Event Snapshot s). To transform this unstructured text into a traversable topology, we leverage the Graphiti framework (Rasmussen et al., 2025). Specifically, we employ an LLM-based extraction module that processes s to identify salient entities (e.g., tool outputs, navigational landmarks) and their semantic relations (e.g., *part_of*, *contradicts*). These extracted elements are instantiated as nodes and edges in the Knowledge Graph $\mathcal{G}_t$. Crucially, the original high-fidelity snapshot s is stored as a property of the central event node, ensuring that the graph retains both the structural skeleton for retrieval and the atomic details for reading.

3.3 Introspective Retrieval via Thought-Driven Navigation

While consolidation resolves the storage limit, the utility of the scaffold lies in its retrieval dynamics. Unlike standard RAG systems that passively wait for user queries, our agent employs an Introspective Retrieval Mechanism to proactively address reasoning gaps.

Thought-as-Query Trigger. The retrieval process is initiated endogenously by the agent's internal Chain-of-Thought. Instead of relying on external prompts, we treat the current reasoning trace R_t generated by π_θ as a latent query source. The system continuously monitors R_t for informational

gaps, converting the immediate thought stream into an introspective query vector $q_{int} = \text{Embed}(R_t)$ without requiring explicit user intervention.

Dual-Path Navigation. Upon triggering, the system executes a parallelized "Dual-Path" strategy to recover structure from the Knowledge Graph $\mathcal{G}_t$: (i) The Entity Path (Semantic Recall) targets discrete Event Nodes $v \in \mathcal{G}_t$ via a hybrid search. A node is retrieved if its content similarity exceeds a threshold: $\text{sim}(q_{int}, \text{Embed}(v)) > \delta$, augmented by sparse BM25 matching to mitigate semantic drift. (ii) The Relation Path (Structural Expansion) simultaneously retrieves Relational Edges $e \in \mathcal{G}_t$ connected to the identified entities. Crucially, this stream applies Topological Reranking (via Graph Distance minimization) to prioritize facts that are structurally proximal, thereby recovering the logical skeleton (e.g., $\xrightarrow{\text{CAUSED\_BY}}$ chains) surrounding isolated facts.

Context Augmentation. To maintain the stability of the reasoning stream, retrieved information is integrated naturally as an observation. Retrieval operates asynchronously and does not interrupt the generation of the reasoning trace R_t. Instead, the synthesized sub-graphs (Entities V_{ret} + Relations E_{ret}) are formatted as a structured observation block OBS_{ret}. As formally defined in Eq. (3), this observation is appended to the current interaction context. This mechanism closes the cognitive loop: Thinking triggers Retrieval $\rightarrow$ Navigation synthesizes Structure $\rightarrow$ Context Augmentation refines future Thinking.

4 Experiments

In this section, we empirically evaluate the proposed Cognitive Scaffold framework. Our experiments are designed to assess the system's performance on long-horizon tasks, the fidelity of our compression mechanism, and the computational efficiency of the decoupled architecture.

Implement. Our framework is implemented using a decoupled inference architecture to optimize the trade-off between reasoning capability and computational cost: (i) Reasoning Model (π_θ): The reasoning backbone is instantiated with Tongyi-DeepResearch-30B-A3B(Team et al., 2025b) as a centralized planner for multi-step tool-augmented reasoning. During inference, we adopt nucleus sampling with (`temperature` $= 0.5$, `top_p` $= 0.9$) and enable `logprobs` to support confidence-aware decision making and trajectory scoring. To control computational budget, we cap the generation length with `max_tokens` = 8192 per call. We further apply `presence_penalty` = 1.1 to suppress degenerate repetitions and encourage exploration over diverse hypotheses. (ii) Compression Model ($\mathcal{M}_\phi$): The consolidation module is initialized with Qwen3-8B (Yang et al., 2025) and fine-tuned on the synthesized RFT dataset using LoRA (Rank = 64, $\alpha = 16$), enabling lightweight deployment while preserving high fidelity of numerical values and entities. The system adopts a base threshold $\tau_{\text{base}} = 10 \times 1024$, an incremental step $\Delta\tau = 3 \times 1024$, and $\eta = 3$ to regulate the frequency of compression and context expansion. (iii) Knowledge Graph: The Knowledge Graph (Rasmussen et al., 2025) is managed via Neo4j.Hardware details are provided in Appendix G.

Benchmarks. We evaluate on three diverse datasets designed to stress-test long-term context capabilities: (i) Xbench-DeepSearch(Team, 2025b): a comprehensive benchmark requiring the agent to synthesize information over long interaction trajectories to answer complex queries. (ii) BrowseComp-ZH(Zhou et al., 2025): real-world Chinese web-browsing tasks, where the agent must navigate noisy HTML pages and maintain goal consistency over many clicks; this dataset is particularly challenging due to the high density of information on each page. (iii) GAIA(Mialon et al., 2023): we use the text-only subset of GAIA questions, evaluating the agent's ability to perform multi-step reasoning and answer short, factual queries without relying on external tools or multimodal inputs.

Baselines. We compare against two categories of methods:(i) LLM-based ReAct Agents: Standard agents utilizing proprietary and open-weights LLMs within a linear ReAct loop, serving as baselines for raw reasoning capability under linear context accumulation.(ii) DeepResearch Agents: Specialized systems designed for long-horizon autonomous research, including both commercial black-box services and recent open-source approaches that employ dynamic context management techniques such as periodic summarization or trajectory folding.We report Avg@3 and Pass@3 based on three independent runs per instance: Avg@3 is the average success rate over the three runs, while Pass@3 counts an instance as solved if

Method	Availability	Xbench-DeepSearch		BrowseComp-ZH		GAIA	
		Avg@3	Pass@3	Avg@3	Pass@3	Avg@3	Pass@3
LLM-based ReAct Agent							
GLM-4.5	✗	70.0	–	37.5	–	66.0	–
DeepSeek-V3.1-671B-A37B (DeepSeek Team, 2025)	✓	71.0	–	49.2	–	63.1	–
OpenAI-o3 (OpenAI, 2025b)	✗	66.7	–	58.1	–	70.5	–
OpenAI-o4-mini	✗	–	–	–	–	60.0	–
Claude-4-Sonnet	✗	64.6	–	29.1	–	68.3	–
Kimi-K2-Instruct-1T (Team et al., 2025a)	✓	50.0	–	28.8	–	57.7	–
DeepResearch Agent							
Tongyi DeepResearch(Team et al., 2025b)	✓	73.0	86.0	46.7	63.7	70.9	85.5
OpenAI DeepResearch (OpenAI, 2025a)	✗	–	–	42.9	–	67.4	–
Kimi Researcher (Moonshot AI, 2025)	✗	69.0	–	–	–	–	–
ReSumTool-30B (Wu et al., 2025b)	✓	–	–	–	42.6	–	63.1
AgentFold (Ye et al., 2025)	✓	–	–	47.3	–	67.0	–
Ours	✓	**74.7**	**87.0**	**48.5**	**65.9**	**72.8**	**88.3**

Table 1: **Main Results.** Comparison of Success Rate (%) across three long-horizon benchmarks. **BrowseComp-ZH** evaluates web browsing capabilities in Chinese, while **Xbench** and **GAIA** test general reasoning and tool use. Availability indicates whether the system is publicly released (e.g., code) or closed. Best results among publicly released methods are bolded.

at least one run succeeds.

4.1 Main Performance Comparison

Table 1 shows that Cognitive Scaffold consistently outperforms baselines across all benchmarks. On Xbench-DeepSearch and GAIA, it surpasses leading proprietary agents (e.g., Tongyi DeepResearch), achieving 87.0% and 88.3% Pass@3 respectively. Most notably, on BrowseComp-ZH, it dominates summary-based approaches, achieving 65.9% Pass@3 compared to ReSum's 42.6%. These results confirm that decoupling fluid reasoning from structured memory significantly enhances both stability and retention.

The performance gap is most pronounced on BrowseComp-ZH (+23.3%), a high-noise environment demanding precise navigation. While previous summary-based methods often suffer from semantic blurring—abstracting away critical locators—our RFT-driven crystallization strictly preserves atomic constraints within the Knowledge Graph ($\mathcal{G}_t$). Moreover, by outperforming long-context baselines despite using a bounded window, our results suggest the bottleneck in infinite-horizon reasoning is context purity rather than capacity. By offloading static history to the graph and retrieving it via thought-driven navigation, the framework effectively mitigates the "lost-in-the-middle" phenomenon, ensuring π_θ always operates on a high-signal fluid context. To provide intuition for these quantitative gains, we present a detailed qualitative analysis in Appendix D.

4.2 Analysis of Compression Fidelity

We build 2,500 candidate context–summary pairs (H_{raw}, s) from teacher trajectories (avg. ∼20k-context tokens). After applying the rule-based rewards (format, semantic alignment, compression), we retain 2,200 pairs for training and hold out 300 for testing. With Qwen3-8B as the student compressor, **SFT-All** on all 2,200 pairs slightly increases hallucination (7.6%→8.6%) and reduces entity F1 (75.0→63.9), while **RFT-Filtered** (top 1,500 by composite reward) lowers hallucination to 5.3% and improves F1 to 77.5. The same pattern holds for GLM-4-9B (Hall: 9.4%→6.9%, F1: 74.5→76.3), suggesting rejection sampling on high-reward pairs is key for faithful compression across base LLMs. We compute Hallucination Rate and Entity F1 using an LLM-as-a-judge evaluator; details are in Appendix C. See Appendix G.2 for the training hardware and LoRA setup.

4.3 Efficiency and Scalability

We evaluate efficiency by analyzing the prompting cost of the reasoning model π_θ. For a question with T steps, let $\mathcal{C}_t$ denote the context injected at step t. The cumulative input token cost is defined as $\mathcal{L}_{\text{main}} = \sum_{t=1}^{T} |\mathcal{C}_t|$. Without memory folding, the interaction history grows linearly with the reasoning horizon. Cognitive Scaffold, however, compresses saturated segments into compact snapshots, effectively bounding the active working context.

As shown in Figure 3, on BrowseComp-ZH, the uncompressed baseline consumes 1.08×10^9

Setting	Hallucination ↓	Entity F1 ↑
Qwen3-8B compressor		
Origin (no tuning)	7.6	75.0
SFT-All (2,200 pairs)	8.6	63.9
RFT-Filtered (1,500 pairs)	5.3	77.5
GLM-4-9B compressor		
Origin (no tuning)	9.4	74.5
SFT-All (2,200 pairs)	10.6	65.8
RFT-Filtered (1,500 pairs)	6.9	76.3

Table 2: Compression fidelity on a held-out test set (300 pairs). "RFT-Filtered" denotes rejection-sampling fine-tuning, where only the top-reward 1,500 pairs are used for training.

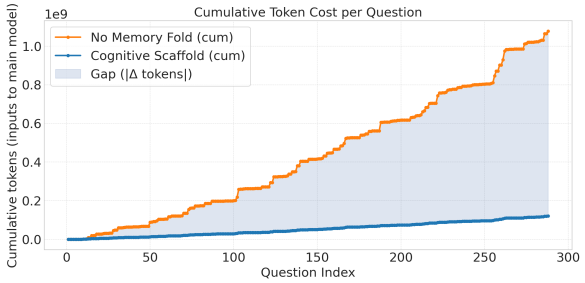

Figure 3: Cumulative main-model token cost on BrowseComp-ZH. The x-axis denotes the question index, and the y-axis shows the cumulative number of input tokens fed into the reasoning backbone π_θ. The shaded area indicates the cumulative token gap between "No Memory Fold" and Cognitive Scaffold.

tokens, while Cognitive Scaffold requires only 1.20×10^8 tokens—an approximately $9\times$ reduction. Compared to state-of-the-art methods like ReSum, which maintains 20k–35k tokens on GAIA and BrowseComp-ZH, our method significantly reduces the average active trajectory size to 12.7k and 13.1k tokens, respectively, representing a reduction of over 50% in static context length.

To provide a comprehensive end-to-end evaluation, we define *Total Tokens* as the aggregate of tokens from the reasoner, compressor, and KG construction. Across BrowseComp-ZH, GAIA, and XBench, the 8B compressor and KG construction (offloaded to a lightweight API) introduce marginal overhead . Even accounting for all components, Cognitive Scaffold maintains a token reduction of $\geq 7.6\times$ over the baseline.

Regarding temporal overhead, we logged the end-to-end wall-clock latency. While structural memory operations and KG construction introduce additional steps, the average completion time per query increases by only 7%–8% across benchmarks. This marginal increase indicates that the significant gains in memory efficiency do not create a computational bottleneck, enabling scalable long-horizon inference. Detailed curves for other benchmarks are provided in Appendix A.

Variant	BrowseComp-ZH		GAIA (text-only)	
	Avg@3	Pass@3	Avg@3	Pass@3
w/o KG retrieval	47.2	64.2	72.3	87.2
w/o RFT compressor	47.9	65.3	71.6	86.4
Full scaffold (ours)	48.5	65.9	72.8	88.3

Table 3: Ablation on scaffold components.

4.4 Ablation on Scaffold Components

We conduct ablation studies on BrowseComp-ZH and GAIA to evaluate the key components of our Cognitive Scaffold, maintaining a consistent reasoning backbone and training data across variants. As shown in Table 3, the full scaffold achieves the best performance, and removing either the KG-based retrieval or the RFT-trained compressor consistently degrades performance.

On our primary large-scale benchmark (BrowseComp-ZH, $N = 288$), these performance drops are statistically significant ($p < 0.05$ after Holm correction). Specifically, removing KG-based retrieval drops the performance from 48.50 ± 1.00 to 47.22 ± 1.59 (Avg@3; $p = 0.0025$), underscoring the importance of structured graph retrieval for re-accessing long-range facts. Similarly, replacing the RFT-trained compressor with the SFT-All variant degrades results to 47.92 ± 0.35 ($p = 0.028$), suggesting that high-fidelity compression is necessary to preserve critical entities and constraints.

On the smaller GAIA dataset ($N = 103$), performance drops follow consistent directions. The degradation without the RFT compressor remains significant prior to correction ($p = 0.045$). While the drop without KG retrieval yields a higher p-value ($p = 0.158$), we attribute this to the limited statistical power inherent in a smaller sample size, as the positive effect direction remains robust across benchmarks. Overall, both structured memory and high-fidelity compression are necessary to fully realize the benefits of the scaffold.

5 Conclusion

We introduce the **Cognitive Scaffold** to resolve the context-noise trade-off in long-horizon agents. By factorizing memory into a fluid context for immediate reasoning and a crystallized Knowledge Graph

for long-term retention, our framework enables precise, hallucination-free synthesis over extended trajectories. While this neuro-symbolic approach significantly outperforms linear baselines, it relies on the fidelity of LLM-based extraction, necessitating robust auditing against error propagation. We open-source our code and trajectories to foster further research into transparent, efficient, and cognitively durable agents.

6 Limitations

Our study has several limitations. First, we evaluate the proposed compression and distillation pipeline under a specific set of tasks and domains. While the design is domain-agnostic in principle, generalization to substantially different domains (e.g., highly technical, legal/medical, or code-heavy corpora) may require re-tuning the compression guideline, reward weights, or format constraints. A broader cross-domain evaluation is left for future work. Second, although our approach is compatible with several adjacent directions, we do not systematically explore these intersections in this paper. For example, integrating privacy- or safety-aware memory filtering, incorporating human-in-the-loop feedback for guideline refinement, or extending the scaffold to multi-agent settings could further improve robustness and applicability.

Acknowledgements

We acknowledge support by the National Science and Technology Major Project of China under grant number 2025ZD1401507, the National Natural Science Foundation of China under Grant 72342004, and the CNPC Innovation Fund 2024DQ02-0503. We also acknowledge the support provided by the High-Performance Computing Platform of Peking University.

References

Sarthak Chandra, Sugandha Sharma, Rishidev Chaudhuri, and Ila Fiete. 2025. Episodic and associative memory from spatial scaffolds in the hippocampus. *Nature*, 638(8051):739–751.

DeepSeek Team. 2025. Introducing deepseek-v3.1: our first step toward the agent era!

Esin Durmus, He He, and Mona Diab. 2020. Feqa: A question answering evaluation framework for faithfulness assessment in abstractive summarization. *arXiv preprint arXiv:2005.03754*.

Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, Dasha Metropolitansky, Robert Osazuwa Ness, and Jonathan Larson. 2024. From local to global: A graph rag approach to query-focused summarization. *arXiv preprint arXiv:2404.16130*.

Runnan Fang, Yuan Liang, Xiaobin Wang, Jialong Wu, Shuofei Qiao, Pengjun Xie, Fei Huang, Huajun Chen, and Ningyu Zhang. 2025. Memp: Exploring agent procedural memory. *Preprint*, arXiv:2508.06433.

Samuel J Gershman, Ila Fiete, and Kazuki Irie. 2025. Key-value memory in the brain. *Neuron*, 113(11):1694–1707.

Mengkang Hu, Tianxing Chen, Qiguang Chen, Yao Mu, Wenqi Shao, and Ping Luo. 2025. Hiagent: Hierarchical working memory management for solving long-horizon agent tasks with large language model. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 32779–32798.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Grégoire Mialon, Clémentine Fourrier, Thomas Wolf, Yann LeCun, and Thomas Scialom. 2023. Gaia: a benchmark for general ai assistants. In *The Twelfth International Conference on Learning Representations*.

Moonshot AI. 2025. Kimi-researcher: End-to-end rl training for emerging agentic capabilities. Blog post. Accessed: 2025-12-25.

Jiayan Nan, Wenquan Ma, Wenlong Wu, and Yize Chen. 2025. Nemori: Self-organizing agent memory inspired by cognitive science. *arXiv preprint arXiv:2508.03341*.

OpenAI. 2025a. Deep research system card.

OpenAI. 2025b. Introducing openai o3 and o4-mini.

Ofir Press, Noah A Smith, and Mike Lewis. 2021. Train short, test long: Attention with linear biases enables input length extrapolation. *arXiv preprint arXiv:2108.12409*.

Zile Qiao, Guoxin Chen, Xuanzhong Chen, Donglei Yu, Wenbiao Yin, Xinyu Wang, Zhen Zhang, Baixuan Li, Huifeng Yin, Kuan Li, and 1 others. 2025. Webresearcher: Unleashing unbounded reasoning capability in long-horizon agents. *arXiv preprint arXiv:2509.13309*.

Preston Rasmussen, Pavlo Paliychuk, Travis Beauvais, Jack Ryan, and Daniel Chalef. 2025. Zep: a temporal knowledge graph architecture for agent memory. *arXiv preprint arXiv:2501.13956*.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. Toolformer: Language models can teach themselves to use tools. *Advances in Neural Information Processing Systems*, 36:68539–68551.

Weiwei Sun, Miao Lu, Zhan Ling, Kang Liu, Xuesong Yao, Yiming Yang, and Jiecao Chen. 2025. Scaling long-horizon llm agent via context-folding. *arXiv preprint arXiv:2510.11967*.

Google Gemini Team. 2025a. Gemini deep research.

Kimi Team, Yifan Bai, Yiping Bao, Guanduo Chen, Jiahao Chen, Ningxin Chen, Ruijue Chen, Yanru Chen, Yuankun Chen, Yutian Chen, and 1 others. 2025a. Kimi k2: Open agentic intelligence. *arXiv preprint arXiv:2507.20534*.

Tongyi DeepResearch Team, Baixuan Li, Bo Zhang, Dingchu Zhang, Fei Huang, Guangyu Li, Guoxin Chen, Huifeng Yin, Jialong Wu, Jingren Zhou, Kuan Li, Liangcai Su, Litu Ou, Liwen Zhang, Pengjun Xie, Rui Ye, Wenbiao Yin, Xinmiao Yu, Xinyu Wang, and 38 others. 2025b. Tongyi deepresearch technical report. *Preprint*, arXiv:2510.24701.

Xbench Team. 2025b. Xbench-deepsearch.

Guanzhi Wang, Yuqi Xie, Yunfan Jiang, Ajay Mandlekar, Chaowei Xiao, Yuke Zhu, Linxi Fan, and Anima Anandkumar. 2023. Voyager: An open-ended embodied agent with large language models. *arXiv preprint arXiv:2305.16291*.

Jialong Wu, Baixuan Li, Runnan Fang, Wenbiao Yin, Liwen Zhang, Zhengwei Tao, Dingchu Zhang, Zekun Xi, Gang Fu, Yong Jiang, and 1 others. 2025a. Webdancer: Towards autonomous information seeking agency. *arXiv preprint arXiv:2505.22648*.

Xixi Wu, Kuan Li, Yida Zhao, Liwen Zhang, Litu Ou, Huifeng Yin, Zhongwang Zhang, Xinmiao Yu, Dingchu Zhang, Yong Jiang, and 1 others. 2025b. Resum: Unlocking long-horizon search intelligence via context summarization. *arXiv preprint arXiv:2509.13313*.

Sikuan Yan, Xiufeng Yang, Zuchao Huang, Ercong Nie, Zifeng Ding, Zonggen Li, Xiaowen Ma, Kristian Kersting, Jeff Z Pan, Hinrich Schütze, and 1 others. 2025. Memory-r1: Enhancing large language model agents to manage and utilize memories via reinforcement learning. *arXiv preprint arXiv:2508.19828*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. 2022. React: Synergizing reasoning and acting in language models. In *The eleventh international conference on learning representations*.

Rui Ye, Zhongwang Zhang, Kuan Li, Huifeng Yin, Zhengwei Tao, Yida Zhao, Liangcai Su, Liwen Zhang, Zile Qiao, Xinyu Wang, and 1 others. 2025. Agentfold: Long-horizon web agents with proactive context management. *arXiv preprint arXiv:2510.24699*.

Howard Yen, Ashwin Paranjape, Mengzhou Xia, Thejas Venkatesh, Jack Hessel, Danqi Chen, and Yuhao Zhang. 2025. Lost in the maze: Overcoming context limitations in long-horizon agentic search. *arXiv preprint arXiv:2510.18939*.

Gongrui Zhang, Jialiang Zhu, Ruiqi Yang, Kai Qiu, Miaosen Zhang, Zhirong Wu, Qi Dai, Bei Liu, Chong Luo, Zhengyuan Yang, and 1 others. 2025a. Infoagent: Advancing autonomous information-seeking agents. *arXiv preprint arXiv:2509.25189*.

Wentao Zhang, Liang Zeng, Yuzhen Xiao, Yongcong Li, Ce Cui, Yilei Zhao, Rui Hu, Yang Liu, Yahui Zhou, and Bo An. 2025b. Agentorchestra: Orchestrating hierarchical multi-agent intelligence with the tool-environment-agent(tea) protocol. *Preprint*, arXiv:2506.12508.

Peilin Zhou, Bruce Leon, Xiang Ying, Can Zhang, Yifan Shao, Qichen Ye, Dading Chong, Zhiling Jin, Chenxuan Xie, Meng Cao, and 1 others. 2025. Browsecomp-zh: Benchmarking web browsing ability of large language models in chinese. *arXiv preprint arXiv:2504.19314*.

A Token Efficiency Analysis on GAIA and XBench

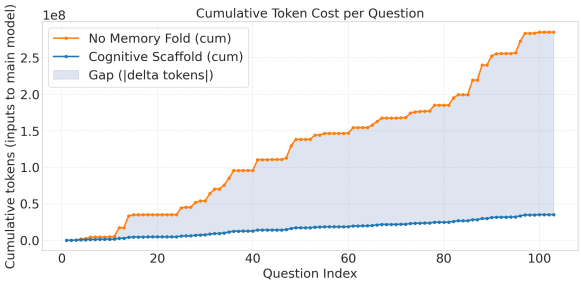

Figure 4: Cumulative main-model token cost on GAIA. The x-axis denotes the question index, and the y-axis shows the cumulative number of input tokens fed into the reasoning backbone π_θ. The shaded area indicates the cumulative token gap between "No Memory Fold" and Cognitive Scaffold.

Token efficiency is evaluated by measuring the cumulative number of input tokens fed into the main reasoning backbone π_θ across questions. As shown in Figure 4 and Figure 5, Cognitive Scaffold consistently reduces the growth rate of cumulative token consumption compared with the "No Memory Fold" setting, and the shaded region visualizes

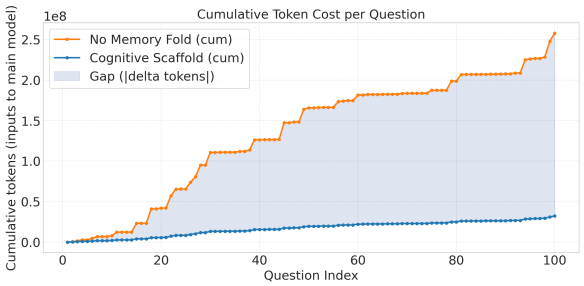

Figure 5: Cumulative main-model token cost on XBench. The x-axis denotes the question index, and the y-axis shows the cumulative number of input tokens fed into the reasoning backbone π_θ. The shaded area indicates the cumulative token gap between "No Memory Fold" and Cognitive Scaffold.

the cumulative token gap between the two variants. This indicates that memory folding and graph-based retrieval can preserve long-horizon reasoning capability while keeping the visible context compact, leading to substantially lower main-model token cost on both benchmarks.

B Pseudocode of the Cognitive Scaffold Inference Framework

This appendix provides the pseudocode of our Cognitive Scaffold Inference Framework. As shown in Algorithm 1, the framework performs tool-augmented reasoning under a strict context budget by maintaining a working context that accumulates model generations and tool feedback, optionally retrieving event snapshots from the constructed knowledge graph. When the context length exceeds a token threshold, it triggers a folding operation via a compression model to summarize selected historical interaction rounds, stores the summary as a new node in the knowledge graph, and removes the folded content from the visible context while preserving only the most recent raw rounds. This design balances information retention (through structured memory in the graph) and context efficiency (through compression), enabling long-horizon reasoning and producing the final answer from the maintained reasoning trace.

C Automatic Evaluation of Compression Fidelity

We automatically evaluate compression fidelity for each context–summary pair (H_{raw}, s) using an LLM-as-a-judge in a strict JSON-only mode. The evaluator is prompted (i) to extract atomic factual claims from s and verify their support in q, and (ii)

Algorithm 1 Cognitive Scaffold Inference Framework

Require: User query q; reasoning model π_θ; compression model $\mathcal{M}_\phi$; knowledge graph $\mathcal{G}_t$; tool set $\mathcal{T}$; token threshold $\tau(k)$; step budget B.
Ensure: Final answer y.

Initialize working context $\mathcal{C}_0 \leftarrow [\texttt{system\_prompt}, q]$
$t \leftarrow 0$
while $t < B$ **do**
  Sample next token/action $a_t \sim \pi_\theta(\cdot \mid \mathcal{C}_t)$
  if <final_answer> is detected in a_t **then**
    Parse y from a_t
    return y
  end if
  if a_t contains a tool call $u \in \mathcal{T}$ **then**
    Execute u and obtain tool response r_t
    Append a_t and r_t to the working context $\mathcal{C}_t$
  end if
  if a_t calls graph_search **then**
    Query $\mathcal{G}_t$ to retrieve relevant event snapshots $\mathcal{S}_t$
    Append $\mathcal{S}_t$ into $\mathcal{C}_t$ as additional context
  end if
  if $\text{len}(\mathcal{C}_t) > \tau(k)$ **then**
    Select a block of past interaction rounds H_t to fold
    $s_t \leftarrow \mathcal{M}_\phi(q_{H_t}, H_t)$ ▷ returns <summary>...</summary>
    Insert s_t as a node into $\mathcal{G}_t$ and update edges
    Mark H_t as folded and remove it from the visible context
    Keep only the last k raw rounds and retrieved snapshots in $\mathcal{C}_t$
  end if
  $t \leftarrow t + 1$
end while
return the best answer extracted from the reasoning history in $\mathcal{C}_t$

to extract salient entities/numbers from both q and s to measure preservation.

Hallucination Rate (claim-level). The judge extracts up to K atomic claims $\{c_i\}_{i=1}^n$ from s (each c_i is a single factual statement). For each claim, it searches for the best supporting evidence span in q; if no supporting span exists, the evidence is marked as NONE. Each claim is labeled as one of: ENTAILED, NOT_SUPPORTED, or CONTRADICTED. We define the hallucination rate as the fraction of claims that are not supported or contradicted by the source:

$$\text{Hall}(q, s) = \frac{1}{n} \sum_{i=1}^{n} \mathbb{I}[\ell_i \in \mathcal{B}], \quad \mathcal{B} = \{\text{NOT\_SUPPORTED}, \text{CONTRADICTED}\}. \tag{6}$$

where ℓ_i is the judge label for claim c_i. We also report the contradiction rate $\text{Contra}(q, s) = \#\{i : \ell_i = \text{CONTRADICTED}\}/n$.

Entity F1 (salient entity/number preservation). The judge extracts up to M salient entities/numbers/constraints from q as a gold list G and up to M from s as a predicted list P. Salient items include PERSON/ORG, model names, locations, dates, numbers/thresholds, file paths, URLs, and method names. We normalize surface forms by lowercasing, collapsing whitespace, and removing thousand separators in numbers, then deduplicate each list. We perform greedy one-to-one matching between P and G using exact match, optionally enabling fuzzy string similarity (SequenceMatcher ratio) with threshold τ. Let TP be the number of matched items, $\mathrm{FP} = |P| - \mathrm{TP}$, and $\mathrm{FN} = |G| - \mathrm{TP}$. Precision, recall, and F1 are:

$$\begin{aligned} \mathrm{Prec} &= \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FP}}, \\ \mathrm{Rec} &= \frac{\mathrm{TP}}{\mathrm{TP}+\mathrm{FN}}, \\ \mathrm{F1} &= \frac{2\,\mathrm{Prec}\,\mathrm{Rec}}{\mathrm{Prec}+\mathrm{Rec}}. \end{aligned} \tag{7}$$

In our implementation we use K=10, M=20, and τ=0.92 when fuzzy matching is enabled.

D Case Study: Resolving Contextual Noise

We analyze a real-world multi-hop query requiring the agent to identify a restaurant near *Fengchao Theater* (inferred from a quote) that is encountered while "going downstairs" and is "associated with the local area" (Dongzhimen).

Baseline Failure. Linear agents efficiently retrieve the theater but falter under noisy HTML listing restaurants on multiple floors (B1, 1F, 2F). Lacking structural grounding, they rely on heuristics—either selecting the ground floor (*Huoban BBQ*, 1F) or matching keywords like "downstairs" (*Onyasai*, B1)—while crucially ignoring the "area-related" semantic constraint.

Scaffold Success. Our framework identifies the correct target, *Dongzhimen Shuanrou* (2F), through two crystallized mechanisms (details in App. E and F): (1) **Aggregated View:** The compression model consolidates the floor plan into a single node summary (**Node 3**), clarifying the descent topology ($2F \rightarrow 1F \rightarrow B1$). (2) **Semantic Anchoring:** It retrieves **Fact 7**, which explicitly links the restaurant to the "Dongzhimen" area. This retrieved fact acts as a decisive filter, overriding spatial heuristics to validate the 2nd-floor option.

Factor	Base-A	Base-B	Ours
Spatial	1F (Heuristic)	B1 (Keyword)	2F (Topology)
Semantic	Ignored	Ignored	Fact 7 Match
Result	*Huoban* ✗	*Onyasai* ✗	*Shuanrou* ✓

Table 4: Reasoning comparison. Baselines fail due to spatial heuristics (1F/B1) and semantic blindness, while Ours succeeds via topology and explicit fact retrieval.

E Detailed Case Study Analysis

In this section, we provide the raw evidence referenced in Section D, including the crystallized knowledge graph nodes, the compression summary, and a step-by-step comparison of the agent's reasoning process.

E.1 Constructed Cognitive Scaffold (KG Snapshot)

Table 5 presents the actual entity nodes and facts instantiated by our system during the exploration of the "Fengchao Theater" trajectory. These structured records directly support the spatial and semantic reasoning described in the main text.

E.2 Crystallized Compression Summary

Table 6 summarizes the key failure modes of the baselines versus the successful mechanism of our method.

The following text represents the actual output from the Compression Model ($\mathcal{M}_\phi$), which served as the working memory for the final reasoning step. Note the explicit inference regarding the "area-related" constraint.

> **Compression Summary Output:**
> **User Scenario:** User exited Dongzhimen Station (Line 13), saw a driving school across the street; walked to Fengchao Theater to watch "Rhinoceros in Love" (Quote: "Dusk is when my vision is worst"); saw a restaurant while going downstairs that relates to the local area.
> **Key Information Points:**
>
> 1. Fengchao Theater (No.3 Xinzhong St) is the venue.
> 2. **Floor Layout:** B1 has Onyasai Hotpot; **1F has Huoban BBQ**; **2F has Dongzhimen Shuanrou**.
> 3. Changjian Driving School is 30m from Dongzhimen Station Exit G.
>
> **Inference:** While the exact floor of descent is implicit, **"Dongzhimen Shuanrou" is speculated to be the target** due to its explicit naming association with the "Dongzhimen" area, unlike the other candidates.

ID	Content / Summary
Node 3	**Entity: Fengchao Theater** *Summary:* Located at No.3 Xinzhong Street, near Dongzhimen. It hosts the play "Rhinoceros in Love". **Contains Onyasai Hotpot (B1), Huoban BBQ (1F), and Dongzhimen Shuanrou (2F).**
Node 5	**Entity: Dongzhimen Shuanrou** *Summary:* Located on the **2nd floor of Fengchao Theater**. **Identified as one of the area-related restaurants.**
Fact 7	**Relation:** Dongzhimen Shuanrou is strongly associated with the Dongzhimen area.
Fact 10	**Relation:** Driving schools near Dongzhimen Station include Changjian Driving School (30m from Exit G).

Table 5: Key components of the Cognitive Scaffold instantiated during the task. Note how Node 3 aggregates the floor plan, and Node 5 explicitly encodes the spatial (2F) and semantic (area-related) attributes.

Dimension	Baseline A (Failure)	Baseline B (Failure)	Ours (Success)
Spatial Logic	**Ground-Floor Heuristic** Assumes "downstairs" always targets the 1st floor (Ground).	**Keyword Matching** Matches the word "downstairs" directly to "B1/Underground".	**Topological Deduction** Uses Node 3's summary to deduce descent order: 2F → 1F → B1.
Semantic Filter	**Ignored** Disregards the "area-related" constraint; selects based on popularity.	**Ignored** Misses the semantic link; selects based on text overlap.	**Explicit Match** Retrieves Fact from Node 5: "...is one of the area-related restaurants."
Result	*Huoban BBQ* (1F) ✗	*Onyasai Hotpot* (B1) ✗	*Dongzhimen Shuanrou* (2F) ✓

Table 6: Comparison of reasoning dynamics. The baselines fail due to unstructured context processing (heuristic/keyword bias), while the Scaffold model succeeds by leveraging crystallized spatial topology and explicit semantic facts.

F Detailed Comparison of Reasoning Trajectories

To provide a granular view of how the Cognitive Scaffold resolves the multi-hop spatial ambiguity, we present the translated reasoning traces of the critical decision-making step for the query described in Section D.

F.1 Trajectory 1: Linear Baseline A (Failure - Heuristic Bias)

Model Output (Reasoning Trace):

> "...Based on the search results, the Fengchao Theater has restaurants on multiple levels. The user mentioned 'going downstairs'. Typically, when leaving a building, one goes down to the ground floor. *Huoban Japanese BBQ* is explicitly listed as being on the **1st floor** and is very popular. Therefore, it is the most likely restaurant encountered when reaching the exit level. The user's mention of 'area-related' is likely a general reference to the busy dining atmosphere..."

Final Prediction: Huoban Japanese BBQ (1st Floor). ✗

F.2 Trajectory 2: Linear Baseline B (Failure - Keyword Matching)

Model Output (Reasoning Trace):

> "...I found *Onyasai Hotpot* located on the **B1 floor** (underground). The user's query contains the phrase 'while going down'. The search snippet for Onyasai mentions it is 'located downstairs'. This keyword match suggests Onyasai is the intended target. The other restaurants are on higher floors (1F, 2F), which contradicts the 'downstairs' description if interpreted as a destination..."

Final Prediction: Onyasai Japanese Hotpot (B1 Floor). ✗

F.3 Trajectory 3: Cognitive Scaffold (Success - Structural & Semantic Logic)

Model Output (Reasoning Trace):

"...I need to satisfy three constraints: (1) Near Fengchao Theater, (2) Encountered while going downstairs, (3) Strongly related to the 'Dongzhimen' area. According to the **Instantiated Entity Nodes**:

- Node 3 (Theater) aggregates the layout: 2F (Dongzhimen Shuanrou) $\rightarrow$ 1F (Huoban BBQ) $\rightarrow$ B1 (Onyasai).
- **Spatial Logic:** 'Going downstairs' from the theater implies a descending order. The first facility encountered would be on the **2nd floor**.
- **Semantic Filter:** The user requires an association with the local area. **Node 5** explicitly states: '*Dongzhimen Shuanrou... is one of the area-related restaurants*'. The name itself contains 'Dongzhimen'.

Huoban BBQ (1F) and Onyasai (B1) lack this semantic link in their node summaries. Therefore, despite being on the 2nd floor, *Dongzhimen Shuanrou* is the correct answer."

Final Prediction: Dongzhimen Shuanrou (2nd Floor). ✓

G Hardware Configuration

G.1 DeepResearch-30B-A3B Deployment

We deploy DeepResearch-30B-A3B using vLLM on 8$\times$ NVIDIA A100 GPUs (40GB).

G.2 Compressor Training Setup (SFT/RFT with LoRA)

For the compressor fine-tuning (SFT and RFT), we use 4$\times$ NVIDIA A100 GPUs (40GB) and train with LoRA (Rank = 64, α = 16).

H Prompt Templates

This appendix lists the prompts for compression and automatic evaluation, including the Guideline-Improver and LLM-as-a-judge templates for hallucination and entity preservation. Placeholders in braces (e.g., {q}, {s}) are filled at runtime, and outputs follow strict tag/JSON formats for reproducibility.

Prompt (Guideline-Improver for Memory Compression)

You are an expert instruction designer and memory-compression specialist. Your job is to rewrite and strengthen a guideline prompt that instructs a student model to compress a multi-turn dialogue into a compact, high-fidelity memory note.
The student model sees:

- A fixed system prompt (must NOT be referenced or modified).
- The full multi-turn dialogue history.
- A final user message that contains the compression guideline P (to be improved).

The student must output exactly ONE memory note, wrapped in `<summary>...</summary>`, with no additional commentary, headings, or metadata.

```
[CURRENT_GUIDELINE]
{current_guideline}
[/CURRENT_GUIDELINE]
```

```
[EVALUATION_STATISTICS]
avg_reward: {avg_reward}
avg_similarity_to_reference: {avg_similarity}
avg_length_score: {avg_length_score}
avg_format_score: {avg_format_score}
num_examples: {num_examples}
[/EVALUATION_STATISTICS]
```

```
[EXAMPLES]
{examples_text}
[/EXAMPLES]
```

```
[YOUR TASK]
```
Using the current guideline, the evaluation statistics, and the examples:
1) Diagnose weaknesses of the current guideline:

- Where it is vague or underspecified.
- What critical content it tends to omit or distort.
- Where it causes format violations or verbosity.

2) Produce an improved guideline prompt that makes the student summary:

- Faithful: preserve key facts, numbers, entities, decisions, constraints, and outcomes.
- Tool-aware: include any tool calls, tool outputs/results, and how they influenced decisions.
- Action-oriented: capture TODOs, pending questions, commitments, deadlines (if present), and next steps.
- Structured but compact: prefer short sentences; avoid duplication; keep only what matters for future continuity.
- Non-speculative: do not invent; if uncertain, mark uncertainty briefly (e.g., “unclear/unspecified”).

3) Hard format rules for the student output:

- Output MUST be exactly one `<summary>...</summary>` block.
- No text outside `<summary>...</summary>`.
- No bullet lists unless necessary; if used, keep at most 3–5 bullets.
- Do NOT include analysis, reasoning traces, or meta commentary.

IMPORTANT CONSTRAINTS:

- Do NOT mention or modify the system prompt.
- Do NOT include dataset-specific names, IDs, or private strings.
- Do NOT copy the examples verbatim; generalize the instruction.
- The new guideline must be a SINGLE instruction written in English, suitable as the final user message.

Prompt (Guideline-Improver for Memory Compression)

```
[OUTPUT FORMAT]
```
Output ONLY:
```
[GUIDELINE]
... your improved English guideline here ...
[/GUIDELINE]
```
Do not output anything else.

Table 7: Prompt template (Guideline-Improver for Memory Compression) (continued).

Prompt (LLM-as-a-Judge for Hallucination Evaluation)

[SYSTEM]
`You are a strict evaluator. You MUST output ONLY valid JSON (no markdown, no extra text). If something is unknown, use null or an empty list.`
[USER]
You will evaluate whether the compressed text contains hallucinations.

Definitions:
- Source text Q (ground truth): the original uncompressed history.
- Summary S: the compressed version generated from Q.
- A claim is hallucinated if it is `NOT_SUPPORTED` by Q, or `CONTRADICTED` by Q.

Task:
1) Extract up to `{k_claims}` atomic claims from S. Each claim should be a single factual statement.
2) For each claim, find the best supporting evidence span in Q (a short quote). If none exists, set `evidence = "NONE"`.
3) Label each claim as one of: `"ENTAILED"`, `"NOT_SUPPORTED"`, `"CONTRADICTED"`.
4) Output JSON with fields:
  - `"claims"`: list of objects `{"claim", "label", "evidence"}`
  - `"meta"`: `{"num_claims"}`

Important rules:
- Use `"NOT_SUPPORTED"` if Q does not clearly support the claim (even if it sounds plausible).
- Use `"CONTRADICTED"` only if Q clearly says the opposite.
- Evidence must be copied from Q when label is `ENTAILED` or `CONTRADICTED`; otherwise `"NONE"`.

Q:
```
«<Q_BEGIN
{q}
Q_END»>
```

S:
```
«<S_BEGIN
{s}
S_END»>
```

Table 8: Prompt template (LLM-as-a-judge for claim-level hallucination evaluation).

Prompt (LLM-as-a-Judge for Entity Preservation Evaluation)

[SYSTEM]
`You are a strict evaluator. You MUST output ONLY valid JSON (no markdown, no extra text). If something is unknown, use null or an empty list.`
[USER]
Extract salient entities and numbers for evaluating entity preservation.

Definitions:
- Q is the source text.
- S is the compressed summary.
- "Salient entities" are the minimal set of entities/numbers/constraints that are important for solving downstream tasks, such as: PERSON, ORG, PRODUCT/MODEL, LOCATION, DATE/TIME, NUMBER/THRESHOLD, FILE/PATH, URL, METHOD/ALGO.

Task:
1) From Q, extract up to {max_entities} salient entities into "gold_entities".
2) From S, extract up to {max_entities} entities into "pred_entities".
3) For each entity, provide:
  - "text": the surface form as it appears
  - "type": one of ["PERSON","ORG","MODEL","LOCATION","DATE","NUMBER","FILE","URL","METHOD","OTHER"]

Output JSON:
```
{
 "gold_entities": ["text": "...", "type": "..."],
 "pred_entities": ["text": "...", "type": "..."]
}
```

Q:
```
«<Q_BEGIN
{q}
Q_END»>
```

S:
```
«<S_BEGIN
{s}
S_END»>
```

Table 9: Prompt template (LLM-as-a-judge for salient entity/number preservation evaluation).