Verify Before You Commit: Towards Faithful Reasoning in LLM Agents via Self-Auditing

Wenhao Yuan[1], Chenchen Lin[2], Jian Chen[1], Jinfeng Xu[1], Xuehe Wang[2], Edith Cheuk Han Ngai[1,*]
[1]The University of Hong Kong, [2]Sun Yat-sen University
wenhao.yuan@connect.hku.hk, chngai@eee.hku.hk

Abstract

In large language model (LLM) agents, reasoning trajectories are treated as reliable internal beliefs for guiding actions and updating memory. However, coherent reasoning can still violate logical or evidential constraints, allowing unsupported beliefs repeatedly stored and propagated across decision steps, leading to systematic behavioral drift in long-horizon agentic systems. Most existing strategies rely on the consensus mechanism, conflating agreement with faithfulness. In this paper, inspired by the vulnerability of unfaithful intermediate reasoning trajectories, we propose **S**elf-**A**udited **Ve**rified **R**easoning (SAVER), a novel framework that enforces verification over internal belief states within the agent before action commitment, achieving faithful reasoning. Concretely, we structurally generate persona-based diverse candidate beliefs for selection under a faithfulness-relevant structure space. To achieve reasoning faithfulness, we perform adversarial auditing to localize violations and repair through constraint-guided minimal interventions under verifiable acceptance criteria. Extensive experiments on six benchmark datasets demonstrate that our approach consistently improves reasoning faithfulness while preserving competitive end-task performance.

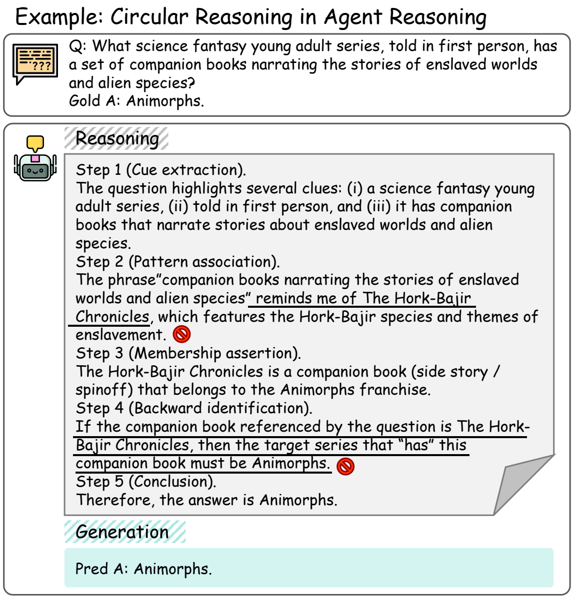

Figure 1: Demonstration of unfaithful agent reasoning. The agent outputs the correct answer '`Animorphs`', but its multi-step reasoning process is logically invalid, as an unverified intermediate assumption "`The phrase ... reminds me of The Hork-Bajir Chronicles`" is used to derive the conclusion that it already presupposes. This failure mode differs fundamentally from unfaithful CoT, where the reasoning is merely an explanatory artifact, while unfaithful reasoning in the agent determines the following behavior and final decision.

1 Introduction

Large Language Models (LLMs) are increasingly deployed as autonomous *agents* that plan, reason, and act over extended horizons. Beyond generating answers, LLM agents maintain internal reasoning trajectories for guiding tool invocation, action commitment, and memory updates across decision steps. With the widespread adoption of reasoning paradigms, such as Chain-of-Thought (CoT) (Wei et al., 2022), trajectories are generally regarded as interpretable representations of the agent's internal state. However, coherent reasoning traces are fragile for decision-making (Lam et al., 2025). Agents may generate seemingly fluent and structured reasoning, yet violate logical or evidential constraints, reflecting a lack of faithful reasoning (Zhao et al., 2025; Xu et al., 2025b). Such violations are difficult to diagnose from final-task success alone, since correct outcomes can arise from chance, redundancy, or downstream correction, masking the underlying reasoning failure (Chang et al., 2025; Kim et al., 2024), as shown in Figure 1. Unlike single-turn Question Answering (QA), where reasoning can be post hoc and disposable, the agent's

[*]Corresponding author.

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 31201–31225
July 2-7, 2026 ©2026 Association for Computational Linguistics

reasoning outputs are repeatedly used, amplified, and written into memory (An et al., 2025; Jiang et al., 2025; Tang et al., 2025). Consequently, unfaithful belief states (e.g., unsupported inferences or hidden assumptions) can propagate, bias decisions, and trigger costly actions in closed-loop agent systems (Chakraborty et al., 2025). The risk is not merely incorrect answers, but systematic behavioral drift driven by unfaithful internal beliefs.

In agentic systems, existing methods have been adopted to manage uncertainty before committing internal reasoning states to actions, such as self-consistency (Wan et al., 2025; Xie et al., 2024), multi-agent debate (Liang et al., 2026, 2024b), which maintain multiple candidate reasoning trajectories and rely on consensus-based aggregation for acted belief determination (Zhang and Xiong, 2025). Nevertheless, they rest on the problematic premise that consensus is faithfulness. In practice, multiple sampled trajectories frequently share the same implicit assumptions or inference templates, resulting in structurally correlated yet unfaithful belief states that are repeatedly selected, further reinforced by majority voting, and committed to memory (Ke et al., 2025). Additionally, most existing methods interact with reasoning at the level of surface text rewriting (Shu et al., 2024), without identifying the logical constraints that the specific reasoning step violates, and verifiable acceptance criteria for committing corrected belief states. These limitations reveal that current LLM agents lack an objective of ensuring reasoning faithfulness before action commitment, raising a key question: *How can an LLM agent verify the reasoning faithfulness without relying on final-task accuracy or consensus?*

To address these challenges, we propose **S**elf-**A**udited **Ve**rified **R**easoning (SAVER), a novel framework for enhancing reasoning faithfulness in LLM agents. Rather than relying on final-task outcomes, SAVER explicitly models the faithfulness of intermediate reasoning steps. To mitigate correlated failure patterns in belief generation, the agent generates a persona-conditioned coalition to elicit structurally diverse candidate belief states and reduce repeated unfaithful templates. It then selects beliefs in a faithfulness-relevant structure space via a quality-aware diversity kernel and k-DPP sampling, followed by adversarial auditing that localizes violations into auditable diagnostics. Finally, SAVER introduces a constraint-guided minimal counterfactual repair protocol that edits only localized failure slices under verifiable acceptance criteria, iterating auditrepair until the belief passes all checks before being committed to actions or memory. Our key contributions are as follows:

- We reveal the overlooked issue of reasoning faithfulness in LLM agents and identify the challenges of verifying intermediate beliefs before action commitment.
- We introduce SAVER, a novel self-auditing framework that verifies and repairs intermediate reasoning trajectories in agents.
- We conduct extensive experiments on multiple public datasets to demonstrate the effectiveness of our approach in improving reasoning faithfulness.

2 Related Work

2.1 Faithful Reasoning in LLMs

LLMs have shown strong performance on reasoning tasks. Existing work has explored the prompting strategies, most notably CoT prompting (Wei et al., 2022; Kojima et al., 2022). Extensions such as program-based reasoning (Chen et al., 2023; Zhang et al., 2024; Jiang et al., 2024; Wang et al., 2024) and least-to-most prompting (Zhou et al., 2023; Arora et al., 2023) further structure these steps and improve task accuracy (Ma et al., 2025). However, improved reasoning performance does not necessarily imply faithful reasoning: empirical studies show that generated chains of thought may rely on spurious signals, and intervening on them has limited impact on model predictions, suggesting that such explanations are post-hoc rationalizations (Lyu et al., 2023; Turpin et al., 2023).

Motivated by these concerns, a growing body of work has focused on evaluating reasoning faithfulness in LLMs (Luo et al., 2024; Paul et al., 2024; Luo et al., 2025). Existing approaches include counterfactual interventions on reasoning traces (Paul et al., 2024; Ding et al., 2024; Joshi et al., 2024), causal probing methods (Chi et al., 2024; Roy et al., 2025), and targeted diagnostics for chain-of-thought faithfulness (Li et al., 2025). Beyond evaluation, several methods improve faithful reasoning through verification (Dougrez-Lewis et al., 2025) or self-consistency (Wan et al., 2025).

Despite these advances, most studies on faithful reasoning focus on LLMs. In agentic settings, however, reasoning trajectories function as persistent belief states to guide downstream decisions,

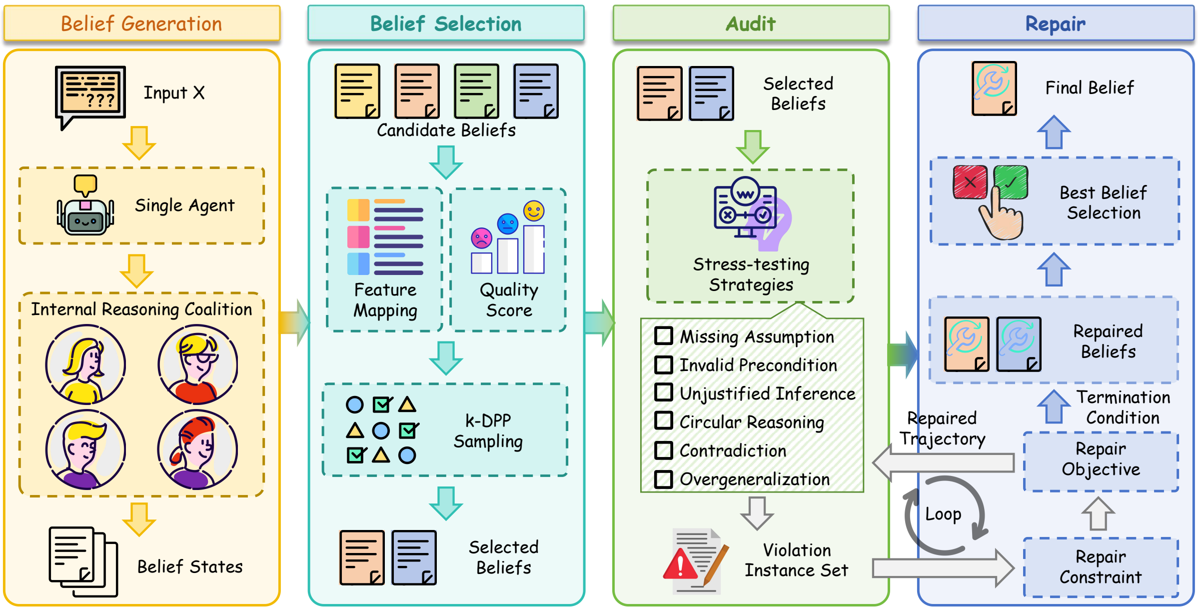

Figure 2: An overview of our proposed SAVeR framework. This diagram illustrates the overall closed-loop workflow, highlighting the end-to-end process from belief generation and selection to auditing and iterative repair.

allowing erroneous beliefs to accumulate, propagate, and trigger costly actions over long horizons. LLM-centric methods to faithful reasoning are insufficient for agentic decision-making, underscoring the need for agent-specific mechanisms.

2.2 Faithfulness-Aware Reasoning in Agents

An emerging research perspective frames LLMs as agents that perform multi-step reasoning through planning, tool use, and interaction with external environments (Hong et al., 2025; Chen et al., 2025; Xu et al., 2025a). These frameworks expose explicit reasoning trajectories to support complex decision-making in interactive settings (Yao et al., 2023; Yang et al., 2024). However, explicit reasoning does not guarantee faithfulness: empirical studies demonstrate that agent-generated plans or tool-use rationales may not causally determine outcomes, but instead serve as plausible post-hoc justifications (Barez et al., 2025).

Despite these observations, existing approaches for mitigating unfaithful agent reasoning remain post-hoc or outcome-driven. Many methods improve robustness by sampling multiple reasoning trajectories or applying external critics, without explicitly verifying whether intermediate reasoning steps are supported by the agents available evidence at the time of decision-making (Liang et al., 2024a; Kostka and Chudziak, 2025). Moreover, numerous approaches operate at surface-level trajectory rewriting or consensus aggregation, which is insufficient to identify structurally correlated yet unsupported belief states that can repeatedly pass heuristic checks (Fu et al., 2025; Grötschla et al., 2025). Consequently, current agentic frameworks lack a mechanism for auditing and verifying internal belief states before action, allowing unfaithful reasoning to be propagated and stored in memory.

3 Methodology

In this section, we introduce the SAVeR framework for faithful reasoning in agentic systems. The complete workflow is shown in Figure 2. We first formalize reasoning faithfulness in § 3.1, and then describe persona-conditioned belief generation in § 3.2 and structure-aware belief selection in § 3.3. We present our reasoning audit mechanism in § 3.4 and introduce the repair procedure that iteratively corrects unfaithful beliefs in § 3.5.

3.1 Modeling Faithful Reasoning in Agent

We consider an agent that generates a multi-step internal reasoning trajectory and then commits to external actions. While improving interpretability, it also introduces the challenge of *reasoning faithfulness*: intermediate reasoning steps may not be fully supported by the information available during decision-making, motivating an explicit formulation of reasoning faithfulness.

Given an input task x, we model the agent's internal reasoning process as a sequence of discrete steps $\tau = (s_1, \ldots, s_L)$, where L denotes the length of the reasoning trajectory and each step s_l represents a local claim, inference, assumption, or intermediate conclusion produced by the agent. To quantify whether a reasoning step is justified under the information available to the agent, we introduce the *support function* $\Gamma(s_l \mid x, \mathcal{H}_l, \mathcal{E}_l) \in [0,1]$, where the reasoning history $\mathcal{H}_l = (s_1, \ldots, s_{l-1})$ and $\mathcal{E}_l$ represents the accessible evidence at step l, including retrieved documents, tool outputs, or environment observations. Thus, we define the *trajectory-level unfaithfulness rate* as

$$U(\tau) = \frac{1}{L}\sum_{l=1}^{L} \mathbb{I}[\Gamma(s_l \mid x, \mathcal{H}_l, \mathcal{E}_l) < \epsilon], \quad (1)$$

where $\mathbb{I}[\cdot]$ denotes the indicator function and a reasoning step is considered *unfaithful* if its support score falls below a predefined threshold ϵ.

3.2 Persona-Conditioned Belief Generation

Unfaithful reasoning in agents typically arises from repeatable and structurally correlated failure patterns, making naively sampled reasoning traces unreliable. Rather than improving final answer accuracy by generating more traces, our goal is to promote *structural diversity* among candidate belief states, so that distinct reasoning modes and failure triggers are explicitly exposed.

For the given input x, we instantiate an *internal reasoning coalition*[1] consisting of M reasoning personas by $\mathcal{A} = \{a_1, a_2, \ldots, a_M\}$, which models a *single* LLM agents internal cognitive diversity, where each persona corresponds to a distinct *structural reasoning bias*, e.g., assumption-first vs. evidence-first. Each persona a_i is instantiated via persona-specific instruction constraints and reasoning templates. Let $\mathcal{Y} = \mathcal{C} \times \mathcal{R}$ denote the belief space, where $\mathcal{C}$ is the claim space and $\mathcal{R}$ is the space of reasoning trajectories. Persona a_i then produces belief $y_i = G(x; a_i) \in \mathcal{Y}$. We denote $y_i = (c_i, r_i)$, where the persona's final claim or decision $c_i \in \mathcal{C}$ and $r_i = \{s_{i,1}, \ldots, s_{i,L_i}\} \in \mathcal{R}$ denotes the full reasoning trajectory with L_i steps and is treated as a *candidate belief state*.

3.3 Structure-Aware Belief Selection

To select diverse subsets of belief states, we define a structural feature mapping $\phi : r_i \mapsto \phi(r_i) \in \mathbb{R}^d$, designed as a proxy for reasoning faithfulness-relevant structure. We decompose it as

$$\phi(r_i) = [g(r_i), p(r_i), v(r_i), s(r_i)]^\top, \quad (2)$$

where *granularity features* $g(r_i)$ quantify step granularity and potential skipping risk; *assumptive features* $p(r_i)$ reflect how assumptions are introduced and managed, capturing missing/implicit premises and whether assumptions are properly scoped; *verification features* $v(r_i)$ measure verification behaviors; *structural-type features* $s(r_i)$ describe the global organization of reasoning.

We introduce a lightweight quality scoring function $q(y_i; x)$, which provides coarse filtering of candidate belief states and removes traces that violate minimal usability constraints (e.g., nonsensical steps or internally inconsistent conclusions). Then, we define a *quality-aware diversity kernel* matrix $I \in \mathbb{R}^{M \times M}$ with entries

$$I_{ij} = \exp(\beta\, \tilde{q}_i) \exp(\beta \tilde{q}_j) \kappa(\phi(r_i), \phi(r_j)), \quad (3)$$

where β controls the quality weighting strength, $\tilde{q}_i$ denotes a normalized $q(y_i; x)$, and $\kappa(\cdot, \cdot)$ is a structural similarity kernel applied to $\phi(r_i)$. Given candidate reasoning outputs generated by M personas, the agent selects K belief states for auditing. We adopt a k-Determinantal Point Process (k-DPP) to sample a subset S of size K:

$$\mathbb{P}(S) \propto \det(I_S), S \subseteq \{1, \ldots, M\}, \quad (4)$$

where I_S denotes the principal submatrix of I defined in Eq. (3) indexed by S. The determinant $\det(I_S)$ favors subsets with structurally complementary belief representations in the induced feature space, thereby encouraging the selection of beliefs that exhibit distinct reasoning patterns. As a result, the sampled set avoids allocating auditing capacity to multiple beliefs that violate reasoning faithfulness in similar ways, and instead increases coverage over diverse *unfaithful reasoning* modes.

3.4 Adversarial Reasoning Audit

Diversity alone does not guarantee reasoning faithfulness, as beliefs may still contain logically unsupported or unjustified steps. To prevent unfaithful beliefs from being committed to actions, we introduce an *adversarial reasoning auditing* procedure to examine belief states, identify faithfulness-violating reasoning steps, and produce structured diagnostic signals for subsequent repair. Notably,

[1]The internal reasoning coalition refers to a set of prompt-conditioned personas instantiated within the same underlying model, where each persona corresponds to a different prompting constraint or reasoning perspective. All candidates are generated by a single LLM, and no inter-agent communication or independent parameterization is involved.

the auditor interrogates the belief state rather than generating or aggregating alternative answers.

We perform reasoning auditing by applying a collection of complementary stress-testing strategies to each selected belief y_i, $i \in S$. Given the reasoning trajectory r_i and input x, the auditor operates under an observable context that aggregates stated assumptions, verified intermediate facts, and admissible evidence (e.g., retrieved documents or tool outputs). Each stress-testing strategy audits r_i from distinct logical perspectives and produces structured audit evidence following a fixed schema to ensure auditability and comparability across beliefs (detailed in Appendix A.2). According to the auditing outcomes, we categorize the faithfulness violations into a type set: $\mathcal{T} = \{$`Missing_Assumption`, `Invalid_Precondition`, `Unjustified_Inference`, `Circular_Reasoning`, `Contradiction`, `Overgeneralization`$\}$, which captures the common failure modes in agentic settings, thereby enabling downstream corrective actions (detailed in Appendix A.1).

For each belief trajectory r_i, the auditor outputs a violation instance set $\mathcal{V}(r_i) = \{(t_{i,j}, l_{i,j})\}_{j=1}^{m_i}$, where $m_i = |\mathcal{V}(r_i)|$ denotes the number of violation instances detected in trajectory r_i, $t_{i,j} \in \mathcal{T}$ denotes the faithfulness violation type, and $l_{i,j} \in \{1, \ldots, L_i\}$ indexes the reasoning step at which the violation is triggered, distinguishing between globally unfaithful beliefs and those that fail only at specific steps. The auditing process operationalizes this notion by identifying steps $s_{i,l}$ for which $\Gamma(s_{i,l} \mid x, \mathcal{H}_{i,l}, \mathcal{E}_{i,l}) < \epsilon$ and mapping each instance to a concrete violation type $t \in \mathcal{T}$. In this way, the violation instance set $\mathcal{V}(r_i)$ can be viewed as a discrete, structured instantiation of support assessment. To represent the belief's faithfulness failure characteristics, we summarize the auditing outcome as an unfaithfulness profile $\mathbf{h}(r_i) = [h_t(r_i)]_{t\in\mathcal{T}}$, where $h_t(r_i)$ counts the number of violations of type t in trajectory r_i.

3.5 Constraint-Guided Belief Repair

Auditing alone does not improve reasoning faithfulness, and full regeneration of new reasoning traces will generally break the causal link between critique and correction, and make it hard to guarantee the removal of originally identified failure. Consequently, we adopt a *minimal counterfactual intervention* principle that only edits the localized failure slices identified, while keeping unaffected steps stable to preserve auditability and prevent unnecessary drift. Specifically, for each violation instance $(t_{i,j}, l_{i,j}) \in \mathcal{V}(r_i)$, the auditor returns structured evidence and an acceptance criterion in a fixed schema (detailed in Appendix A.2), converting subjective critique into faithfulness constraints through explicit acceptance criteria (see Appendix B). Given the audit output $\mathcal{V}(r_i)$, we define a repair constraint set $\Theta_i = \{\theta_{i,1}, \ldots, \theta_{i,m_i}\}$, where each constraint $\theta_{i,j}$ encodes a prescribed correction and an explicit criterion for verifying. Let r_i denote the original belief-specific reasoning trajectory to be repaired, and $\tilde{r}_i$ the repaired trajectory, computed by solving

$$\tilde{r}_i = \arg\min_{r} \mathcal{L}_{\text{cons}}(r; \Theta_i) + \lambda\Delta(r, r_i), \tag{5}$$

where the constraint violation loss $\mathcal{L}_{\text{cons}}(r; \Theta_i) = \sum_{j=1}^{m_i} \mathbb{1}[\neg \mathsf{Sat}(r, \theta_{i,j})]$ penalizes failures to satisfy the acceptance criteria implied by Θ_i, $\mathsf{Sat}(r, \theta)$ encodes a concrete and verifiable condition specifying when a violation is resolved, and the minimal edit cost $\Delta(r, r_i)$ measures the deviation between the repaired and original trajectories, enforcing minimal intervention. Correcting one violation can expose additional latent violations. We thereby iterate auditing and repair until no violation instances remain, i.e., $\mathcal{V}(\tilde{r}_i) = \emptyset$, the repaired belief is then committed to action and update memory, preventing the propagation of unfaithful reasoning in long-horizon agentic decision-making.

After repairing the audited subset $\{y_i\}_{i\in S}$, the agent selects a belief for execution by maximizing the quality score $q(\cdot; x)$ while penalizing residual unfaithfulness reflected by $\mathbf{h}(\cdot)$:

$$i^\star = \arg\max_{i\in S}\Big(q(\tilde{y}_i; x) - \alpha \sum_{t\in\mathcal{T}} w_t h_t(\tilde{r}_i)\Big), \tag{6}$$

where $\tilde{y}_i = (\tilde{c}_i, \tilde{r}_i)$ denotes the repaired belief, w_t is a predefined type-dependent severity weight, and α controls the trade-off between superficial quality and verified reasoning faithfulness.

4 Evaluation

4.1 Experimental Setup

Datasets We evaluate our method on six benchmark datasets with various reasoning settings. **Multi-hop QA** integrates information from multiple sources and performs multi-step reasoning. We adopt three benchmarks for this task: HotpotQA (Yang et al., 2018), 2WikiMHQA (Ho et al., 2020), and MuSiQue (Trivedi et al.,

Models	Methods	HotpotQA		2WikiMHQA		MuSiQue		NQ		Quoref		FEVER
		EM ↑	F1 ↑	EM ↑	F1 ↑	EM ↑	F1 ↑	EM ↑	F1 ↑	EM ↑	F1 ↑	EM ↑
LLaMA-3.1-8B	VANILLA LM	34.8	43.6	39.6	47.3	23.8	34.6	29.5	38.2	29.4	37.7	53.7
	COT	38.3	47.5	44.2	50.7	27.1	36.8	32.7	43.4	33.2	42.3	55.9
	MAD	<u>43.1</u>	51.2	**47.9**	<u>55.4</u>	30.9	<u>40.8</u>	<u>36.6</u>	<u>46.9</u>	36.3	45.2	60.7
	SELF-REFINE	40.8	48.9	46.3	53.3	28.7	37.8	33.5	43.0	34.1	43.6	57.6
	B-2	42.9	<u>51.3</u>	46.7	54.4	<u>31.0</u>	40.6	35.9	44.9	<u>36.7</u>	**46.2**	60.6
	SAVER(Ours)	**43.7**	**52.6**	<u>47.7</u>	**55.5**	**31.8**	**42.5**	**37.1**	**47.8**	**37.2**	<u>45.7</u>	**61.1**
LLaMA-3.2-3B	VANILLA LM	30.6	39.1	31.6	39.4	11.7	20.8	25.1	34.8	24.4	34.6	49.3
	COT	34.4	43.6	35.0	43.6	15.2	23.4	29.5	38.6	29.2	37.8	52.4
	MAD	<u>37.4</u>	<u>45.8</u>	<u>39.8</u>	<u>47.2</u>	<u>18.4</u>	<u>28.1</u>	33.4	42.4	**33.5**	<u>41.2</u>	<u>55.6</u>
	SELF-REFINE	34.9	44.1	36.8	44.5	17.1	26.4	31.2	39.2	29.9	39.4	53.8
	B-2	36.0	45.7	39.9	46.9	18.3	27.5	**34.4**	<u>42.9</u>	32.1	40.7	54.3
	SAVER(Ours)	**38.3**	**47.5**	**40.0**	**48.6**	**18.6**	**28.3**	<u>33.9</u>	**43.8**	<u>33.2</u>	**41.9**	**56.4**
Qwen-2.5-7B	VANILLA LM	33.9	42.3	38.4	46.0	21.9	30.5	28.2	36.9	28.4	36.5	52.7
	COT	38.6	46.6	42.3	49.3	26.3	34.7	33.0	41.7	32.8	42.5	56.3
	MAD	<u>42.5</u>	<u>50.9</u>	46.9	54.8	**30.8**	<u>39.3</u>	<u>36.2</u>	44.6	<u>35.3</u>	**44.8**	60.1
	SELF-REFINE	39.4	48.6	44.1	52.2	27.2	36.4	34.6	43.7	33.8	42.9	58.2
	B-2	41.2	50.6	<u>47.2</u>	<u>55.1</u>	29.1	39.0	35.5	<u>45.3</u>	35.1	43.3	<u>60.7</u>
	SAVER(Ours)	**43.1**	**51.2**	**47.7**	**55.8**	<u>30.6</u>	**39.4**	**36.8**	**45.9**	**35.6**	<u>44.1</u>	**60.9**

Table 1: The overall evaluation results of SAVER and other baseline methods on six benchmarks. The best-performed method is marked by **bold** and the runner-up performing method is marked by <u>underline</u>.

2022). **Evidence-sensitive QA** focuses on answering questions or verifying claims where correctness critically depends on whether sufficient and appropriate evidence supports the conclusion, prone to unsupported assumptions and unjustified inference steps. We consider Natural Questions (NQ) (Karpukhin et al., 2020) and FEVER (Thorne et al., 2018) in this category. **Local reasoning** tasks resolve referential dependencies within a single context, serving as a baseline for evaluating reasoning faithfulness under minimal structural uncertainty. We choose Quoref (Dasigi et al., 2019) for evaluation.

Baselines We compare our method against state-of-the-art baselines. We adopt **Vanilla LM** (Brown et al., 2020) as a direct generation baseline to answer questions without explicit reasoning. To elicit step-by-step reasoning, we include **CoT** (Wei et al., 2022), where the model produces a rationale before answering. We consider deliberation-based inference with **Multi-Agent Debate (MAD)** (Liang et al., 2024b), which aggregates multiple agents' discussions to form the final answer. For iterative self-improvement, we adopt **Self-Refine** (Madaan et al., 2023), where the model alternates between generating and revising based on self-critique. Finally, we include **Best-of-2 (B-2)** (Papineni et al., 2002) to produce two candidate outputs and select the final answer.

Evaluation Metrics We utilize two complementary categories of metrics for evaluation. For **Task-level Performance**, we report Exact Match (EM) and token-level F1, following standard evaluation protocols for QA and verification tasks. For **Reasoning Faithfulness**, as correct final answers do not necessarily imply faithful reasoning, we additionally evaluate faithfulness at the trajectory level. Based on the audit results, we compute the following faithfulness metrics: (i) *Average Violations* (Avg Viol), the mean number of detected faithfulness violations per reasoning trajectory; (ii) *Violation-Free Rate* (VFR), the proportion of trajectories that contain no detected violations; (iii) *Unfaithful Step Rate* (USR), the fraction of reasoning steps within a trajectory that are flagged as unfaithful; (iv) *Post-Repair Residual* (Post-Res) measures the remaining violation rate after the auditrepair procedure is applied.

Implementation Details We conduct our experiments on Qwen 2.5-7B (Bai et al., 2025), LLaMA-3.1-8B, and LLaMA-3.2-3B (Dubey et al., 2024; Touvron et al., 2023). All models are used in the zero-shot inference setting, without task-specific fine-tuning. We default person number $M = 4$, select $K = 2$ candidates for auditing. We define $\beta = 1.0$, $\epsilon = 0.5$. The audit-repair process is iterated for at most 10 rounds. Faithfulness evaluation is performed with the same auditing protocol for all methods. Violation statistics are com-

Methods	HotpotQA				2WikiMHQA				MuSiQue			
	Avg Viol ↓	VFR ↑	Post-Res ↓	USR ↓	Avg Viol ↓	VFR ↑	Post-Res ↓	USR ↓	Avg Viol ↓	VFR ↑	Post-Res ↓	USR ↓
VANILLA LM	2.65	7.43%	–	46.41%	2.83	6.58%	–	53.19%	3.25	5.34%	–	62.63%
COT	1.98	24.89%	–	27.36%	2.21	17.41%	–	32.11%	2.91	13.26%	–	37.58%
MAD	1.33	36.74%	–	23.94%	1.81	32.78%	–	28.82%	2.16	26.17%	–	36.51%
SELF-REFINE	1.48	31.80%	0.18	21.77%	1.93	26.59%	0.23	26.73%	2.03	21.96%	0.32	31.45%
B-2	1.62	28.57%	–	23.61%	2.06	23.83%	–	28.15%	2.24	18.67%	–	34.78%
SAVER(Ours)	0.37	81.36%	0.05	9.12%	0.56	72.34%	0.08	13.84%	0.83	69.38%	0.11	19.73%

Table 2: Reasoning Faithfulness Evaluation on Multi-hop QA Benchmarks under LLaMA-3.1-8B.

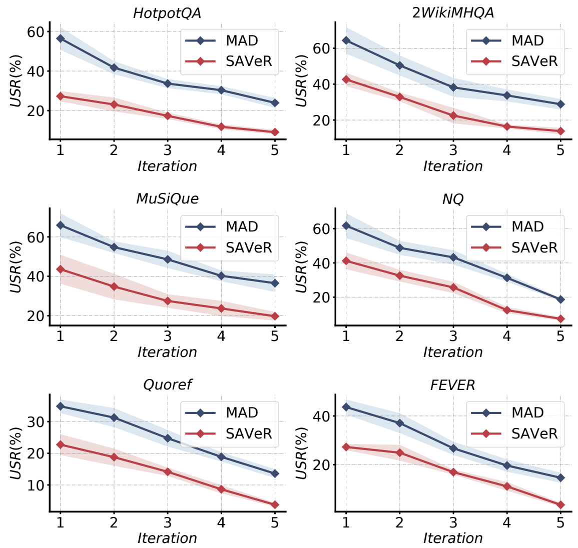

Figure 3: Audit-Repair (A-R) dynamics on HotpotQA. For SAVER, iterations correspond to one audit-repair cycle. For MAD, iterations denote debates to reduce inconsistencies without verifiable acceptance criteria.

puted based on the final reasoning trajectories produced by each method. We run our models on four NVIDIA RTX 4090 GPU devices.

4.2 Main Results

Table 1 reports the overall performance of SAVER and baselines across six benchmarks under three backbone models, where SAVER achieves consistently competitive evaluation results. On multi-hop QA benchmarks (HotpotQA, 2WikiMHQA, and MuSiQue), SAVER demonstrates clear improvements over standard prompting methods and iterative refinement baselines, indicating its effectiveness in handling multi-step reasoning tasks. On evidence-sensitive and single-hop benchmarks (NQ, Quoref, and FEVER), SAVER also performs competitively, suggesting the superiority of enforcing reasoning verification on performance. We further observe that the performance gains of SAVER are stable across different model scales.

We demonstrate the reasoning faithfulness evaluation on three multi-hop QA benchmarks as summarized in Table 2. SAVER consistently achieves substantially lower Avg Viol and USR, along with markedly higher VFR, compared to all baselines across datasets, indicating a significant reduction in unfaithful intermediate reasoning. In contrast, CoT and MAD alleviate unfaithfulness to a limited extent but still retain a large proportion of violation-prone steps. Moreover, the low Post-Res values of SAVER indicate that the audit-repair (A-R) process effectively resolves detected violations.

Figure 3 illustrates the evolution of USR across iterative refinement for SAVER and MAD on six benchmarks under LLaMA-3.1-8B. Across all datasets, SAVER exhibits a faster and more stable reduction in USR, consistently converging to substantially lower unfaithfulness levels than MAD within a small number of iterations. In contrast, while MAD gradually reduces USR through successive debate rounds, a considerable fraction of unfaithful steps persists even after multiple iterations, indicating that explicitly auditing and repairing localized reasoning failures is more effective than debate-based refinement in preventing the accumulation of unfaithful intermediate reasoning.

4.3 Case Studies and Discussions

We present a representative case study to illustrate how unfaithful reasoning arises in agentic question answering and how SAVER mitigates such failures through explicit auditing and repair. As shown in Figure 4, different personas generate diverse belief candidates, including assumption-driven numerical estimation and evidence-first extraction. During auditing, SAVER localizes these failures to specific reasoning slices. In our case, one belief commits a numerical estimate (3,500-4,000 → 3,700) without explicit evidence, which is flagged as an unjustified inference. Another belief identifies the relevant entity correctly but fails to provide a citable sentence linking the arena to its seated capacity, violating required pre-

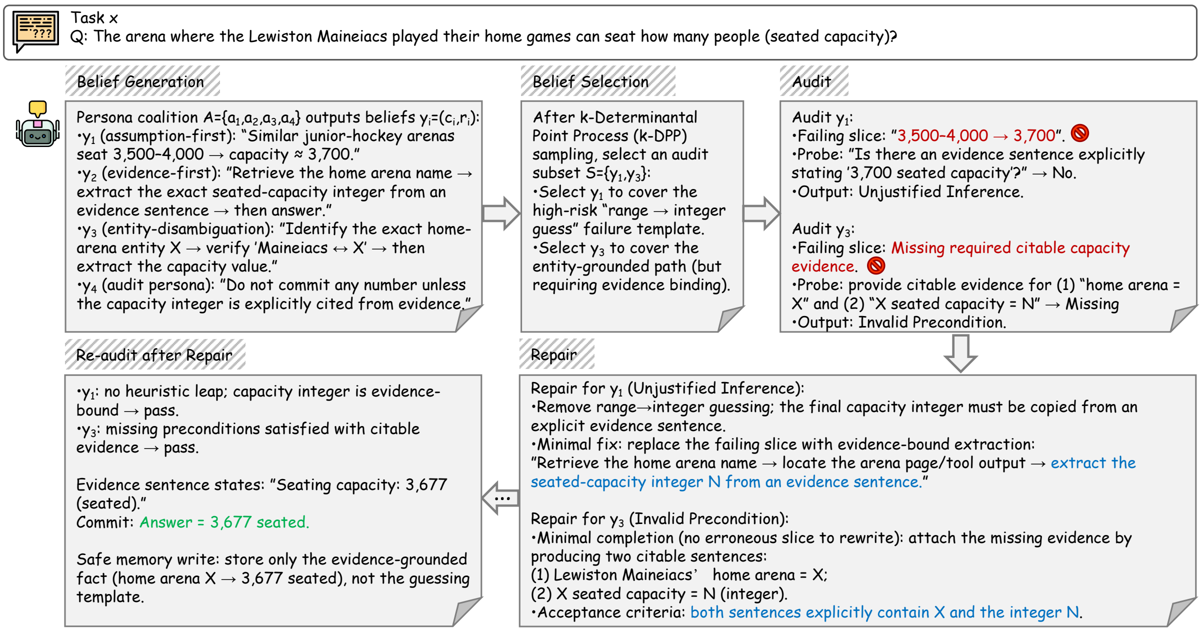

Figure 4: A case study on a multi-hop factual query to correct unjustified inference. The agent initially proposes plausible capacity values based on arena similarity and entity identification, but without explicit evidence. SAVER audits belief candidates, flags unsupported numerical guesses and missing citable capacity statements, and repairs them by enforcing evidence-bound extraction of the exact seated-capacity integer from retrieved sources. After iterative re-auditing, only the verified capacity value is committed to the final answer and written to agent memory.

Methods	HotpotQA						2WikiMHQA					
	EM ↑	F1 ↑	Avg Viol ↓	VFR ↑	Post-Res ↓	USR ↓	EM ↑	F1 ↑	Avg Viol ↓	VFR ↑	Post-Res ↓	USR ↓
w/o Persona	43.2	52.4	0.49	74.55%	0.06	11.97%	47.6	55.3	0.78	65.98%	0.10	19.24%
w/o k-DPP	43.3	52.2	0.64	71.47%	0.08	15.86%	47.5	55.2	0.86	61.78%	0.14	20.52%
w/o Auditing	43.8	52.8	1.37	42.65%	–	26.74%	47.8	55.7	1.76	38.95%	–	29.17%
w/o Repair	44.0	52.9	1.56	33.68%	–	37.63%	48.1	55.6	1.83	29.17%	–	39.84%
SAVER	43.7	52.6	0.37	81.36%	0.05	9.12%	47.7	55.5	0.56	72.34%	0.08	13.84%

Table 3: Ablation Study on the HotpotQA and 2WikiMHQA Benchmarks under LLaMA-3.1-8B.

conditions. Heuristic guessing is replaced with evidence-bound extraction, and missing preconditions are satisfied by explicitly attaching verifiable evidence sentences. The repaired beliefs are then re-audited to ensure all acceptance criteria are met before commitment. As a result, the agent produces a final answer that is fully grounded in cited evidence, preventing the accumulation of unfaithful beliefs in long-horizon reasoning.

4.4 Ablation Studies

In Table 3, we present the ablation study on HotpotQA and 2WikiMHQA, examining the contribution of each component in SAVER. Removing any module consistently degrades reasoning faithfulness, while having marginal effects on EM and F1, indicating the effectiveness on the intermediate reasoning quality. Specifically, removing persona generation leads to a noticeable increase in Avg Viol and USR, suggesting the significance of structured reasoning diversity for exposing distinct failure modes. Disabling the k-DPP-based belief selection further exacerbates unfaithfulness, highlighting the role of structure-aware diversity in preventing correlated reasoning errors. More severe degradation is observed when auditing or repair is removed: both settings result in substantially higher Avg Viol and USR. These results confirm that audit and constraint-guided repair are essential for effectively reducing unfaithful reasoning.

5 Conclusion

In this work, we studied the agent reasoning faithfulness, where coherent reasoning can still violate logical or evidential constraints, and such unfaithful beliefs may propagate and accumulate in agentic systems, leading to systematic behavioral drift. We propose SAVER, a framework that explicitly verifies intermediate belief states before action commitment. SAVER generates diverse candidate beliefs, selectively inspects them at the trajectory level, and corrects localized reasoning failures under explicit acceptance criteria, enabling the agent to prevent unsupported inferences from being written to memory. Extensive experiments across multiple benchmarks demonstrate that SAVER substantially improves reasoning faithfulness while maintaining competitive end-task performance.

Limitations

This work exhibits several limitations worth noting. Firstly, extra computational overhead is introduced by maintaining multiple candidate belief states and performing iterative A-R cycles. Although SAVER limits audit to a small, structurally diverse subset, the A-R loop remains more expensive than single-pass prompting or lightweight refinement strategies, particularly in tasks with short reasoning chains. Secondly, strict faithfulness enforcement may be unnecessary in simple scenarios and could introduce redundant reasoning operations. While SAVER is designed to localize and minimally correct unsupported reasoning steps, it currently lacks an explicit mechanism to adapt verification depth to task difficulty. Future work could explore adaptive auditing policies that condition verification on uncertainty or task complexity, enabling agents to dynamically trade off reasoning faithfulness and efficiency.

Acknowledgments

This work was supported by the UGC General Research Fund no. 17209822 and the Innovation and Technology Commission Fund no. ITS/383/23FP from Hong Kong.

GenAI Usage Disclosure

This work is entirely original and was conducted by the authors. Generative AI tools were not used to produce any content of the work; they were used solely to assist with language refinement and improve clarity and quality of the text.

References

Kaikai An, Fangkai Yang, Liqun Li, Junting Lu, Sitao Cheng, Shuzheng Si, Lu Wang, Pu Zhao, Lele Cao, Qingwei Lin, Saravan Rajmohan, Dongmei Zhang, and Baobao Chang. 2025. Thread: A logic-based data organization paradigm for how-to question answering with retrieval augmented generation. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 18300–18319. Association for Computational Linguistics.

Aseem Arora, Shabbirhussain Bhaisaheb, Harshit Nigam, Manasi Patwardhan, Lovekesh Vig, and Gautam Shroff. 2023. Adapt and decompose: Efficient generalization of text-to-sql via domain adapted least-to-most prompting. In *Proceedings of the 1st GenBench Workshop on (Benchmarking) Generalisation in NLP*, pages 25–47. Association for Computational Linguistics.

Shuai Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, Sibo Song, Kai Dang, Peng Wang, Shijie Wang, Jun Tang, Humen Zhong, Yuanzhi Zhu, Mingkun Yang, Zhaohai Li, Jianqiang Wan, Pengfei Wang, Wei Ding, Zheren Fu, Yiheng Xu, and 8 others. 2025. Qwen2.5-vl technical report. *Preprint*, arXiv:2502.13923.

Fazl Barez, Tung-Yu Wu, Iván Arcuschin, Michael Lan, Vincent Wang, Noah Siegel, Nicolas Collignon, Clement Neo, Isabelle Lee, Alasdair Paren, and 1 others. 2025. Chain-of-thought is not explainability. *Preprint, alphaXiv*, page v1.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Neeloy Chakraborty, Melkior Ornik, and Katherine Driggs-Campbell. 2025. Hallucination detection in foundation models for decision-making: A flexible definition and review of the state of the art. *ACM Computing Surveys*, 57(7):1–35.

Ching Chang, Yidan Shi, Defu Cao, Wei Yang, Jeehyun Hwang, Haixin Wang, Jiacheng Pang, Wei Wang, Yan Liu, Wen-Chih Peng, and Tien-Fu Chen. 2025. A survey of reasoning and agentic systems in time series with large language models. *arXiv preprint arXiv:2509.11575*.

Wenhu Chen, Xueguang Ma, Xinyi Wang, and William W Cohen. 2023. Program of thoughts prompting: Disentangling computation from reasoning for numerical reasoning tasks. *Transactions on Machine Learning Research*.

Zhaoling Chen, Robert Tang, Gangda Deng, Fang Wu, Jialong Wu, Zhiwei Jiang, Viktor Prasanna, Arman Cohan, and Xingyao Wang. 2025. Locagent: Graph-guided llm agents for code localization. In *Proceedings of the 63rd Annual Meeting of the Association* *for Computational Linguistics (Volume 1: Long Papers)*, pages 8697–8727. Association for Computational Linguistics.

Haoang Chi, He Li, Wenjing Yang, Feng Liu, Long Lan, Xiaoguang Ren, Tongliang Liu, and Bo Han. 2024. Unveiling causal reasoning in large language models: Reality or mirage? In *Advances in Neural Information Processing Systems*, volume 37, pages 96640–96670. Curran Associates, Inc.

Pradeep Dasigi, Nelson F Liu, Ana Marasović, Noah A Smith, and Matt Gardner. 2019. Quoref: A reading comprehension dataset with questions requiring coreferential reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 5925–5932, Hong Kong, China. Association for Computational Linguistics.

Bowen Ding, Qingkai Min, Shengkun Ma, Yingjie Li, Linyi Yang, and Yue Zhang. 2024. A rationale-centric counterfactual data augmentation method for cross-document event coreference resolution. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 1112–1140. Association for Computational Linguistics.

John Dougrez-Lewis, Mahmud Elahi Akhter, Federico Ruggeri, Sebastian Löbbers, Yulan He, and Maria Liakata. 2025. Assessing the reasoning capabilities of llms in the context of evidence-based claim verification. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 20604–20628. Association for Computational Linguistics.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, A. Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony S. Hartshorn, Aobo Yang, Archi Mitra, A. Sravankumar, A. Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, and 510 others. 2024. The llama 3 herd of models.

Dayuan Fu, Keqing He, Yejie Wang, Wentao Hong, Zhuoma Gongque, Weihao Zeng, Wei Wang, Jingang Wang, Xunliang Cai, and Weiran Xu. 2025. Agentrefine: Enhancing agent generalization through refinement tuning. In *The Thirteenth International Conference on Learning Representations*.

Florian Grötschla, Luis Müller, Jan Tönshoff, Mikhail Galkin, and Bryan Perozzi. 2025. Agentsnet: Coordination and collaborative reasoning in multi-agent llms. *arXiv preprint arXiv:2507.08616*.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6609–6625, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Sirui Hong, Yizhang Lin, Bang Liu, Bangbang Liu, Binhao Wu, Ceyao Zhang, Danyang Li, Jiaqi Chen, Jiayi Zhang, Jinlin Wang, Li Zhang, Lingyao Zhang, Min Yang, Mingchen Zhuge, Taicheng Guo, Tuo Zhou, Wei Tao, Robert Tang, Xiangtao Lu, and 9 others. 2025. Data interpreter: An LLM agent for data science. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 19796–19821. Association for Computational Linguistics.

Yue Jiang, Qin Chao, Yile Chen, Xiucheng Li, Shuai Liu, and Gao Cong. 2024. UrbanLLM: Autonomous urban activity planning and management with large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 1810–1825. Association for Computational Linguistics.

Zhuohang Jiang, Pangjing Wu, Xu Yuan, Wenqi Fan, and Qing Li. 2025. Qa-dragon: Query-aware dynamic rag system for knowledge-intensive visual question answering. *arXiv preprint arXiv:2508.05197*.

Abhinav Joshi, Areeb Ahmad, and Ashutosh Modi. 2024. Cold: Causal reasoning in closed daily activities. In *Advances in Neural Information Processing Systems*, volume 37, pages 5145–5187. Curran Associates, Inc.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick SH Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781. Association for Computational Linguistics.

Zixuan Ke, Fangkai Jiao, Yifei Ming, Xuan-Phi Nguyen, Austin Xu, Do Xuan Long, Minzhi Li, Chengwei Qin, PeiFeng Wang, silvio savarese, Caiming Xiong, and Shafiq Joty. 2025. A survey of frontiers in LLM reasoning: Inference scaling, learning to reason, and agentic systems. *Transactions on Machine Learning Research*. Survey Certification.

Minsoo Kim, Jongyoon Kim, Jihyuk Kim, and Seungwon Hwang. 2024. QuBE: Question-based belief enhancement for agentic LLM reasoning. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 21403–21423. Association for Computational Linguistics.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. In *Advances in Neural Information Processing Systems*, volume 35, pages 22199–22213. Curran Associates, Inc.

Adam Kostka and JarosĹ Chudziak. 2025. Towards cognitive synergy in llm-based multi-agent systems: integrating theory of mind and critical evaluation. In *Proceedings of the Annual Meeting of the Cognitive Science Society*, volume 47.

Man Ho Lam, Chaozheng Wang, Jen-tse Huang, and Michael Lyu. 2025. Codecrash: Exposing llm fragility to misleading natural language in code reasoning. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, volume 38.

Jiachun Li, Pengfei Cao, Yubo Chen, Jiexin Xu, Huaijun Li, Xiaojian Jiang, Kang Liu, and Jun Zhao. 2025. Towards better chain-of-thought: A reflection on effectiveness and faithfulness. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 10747–10765. Association for Computational Linguistics.

Dayong Liang, Xiao-Yong Wei, and Changmeng Zheng. 2026. Multi-agent undercover gaming: Hallucination removal via counterfactual test for multimodal reasoning. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

Sirui Liang, Baoli Zhang, Jun Zhao, and Kang Liu. 2024a. ABSEval: An agent-based framework for script evaluation. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 12418–12434. Association for Computational Linguistics.

Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Shuming Shi, and Zhaopeng Tu. 2024b. Encouraging divergent thinking in large language models through multi-agent debate. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 17889–17904, Miami, Florida, USA. Association for Computational Linguistics.

Linhao Luo, Yuan-Fang Li, Gholamreza Haffari, and Shirui Pan. 2024. Reasoning on graphs: Faithful and interpretable large language model reasoning. In *The Twelfth International Conference on Learning Representations*.

Linhao Luo, Zicheng Zhao, Gholamreza Haffari, Yuan-Fang Li, Chen Gong, and Shirui Pan. 2025. Graph-constrained reasoning: Faithful reasoning on knowledge graphs with large language models. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pages 41540–41565. PMLR.

Qing Lyu, Shreya Havaldar, Adam Stein, Li Zhang, Delip Rao, Eric Wong, Marianna Apidianaki, and Chris Callison-Burch. 2023. Faithful chain-of-thought reasoning. In *The 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (IJCNLP-AACL 2023)*.

Qianou Ma, Weirui Peng, Chenyang Yang, Hua Shen, Kenneth Koedinger, and Tongshuang Wu. 2025. What should we engineer in prompts? training humans in requirement-driven llm use. *ACM Transactions on Computer-Human Interaction*, 32(4).

Aman Madaan, Niket Tandon, Prakhar Gupta, Skyler Hallinan, Luyu Gao, Sarah Wiegreffe, Uri Alon, Nouha Dziri, Shrimai Prabhumoye, Yiming Yang, Shashank Gupta, Bodhisattwa Prasad Majumder, Katherine Hermann, Sean Welleck, Amir Yazdanbakhsh, and Peter Clark. 2023. Self-refine: Iterative refinement with self-feedback. In *Advances in Neural Information Processing Systems*, volume 36, pages 46534–46594. Curran Associates, Inc.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, USA. Association for Computational Linguistics.

Debjit Paul, Robert West, Antoine Bosselut, and Boi Faltings. 2024. Making reasoning matter: Measuring and improving faithfulness of chain-of-thought reasoning. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 15012–15032. Association for Computational Linguistics.

Amartya Roy, N Devharish, Shreya Ganguly, and Kripabandhu Ghosh. 2025. Causal-LLM: A unified one-shot framework for prompt- and data-driven causal graph discovery. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 8259–8279. Association for Computational Linguistics.

Lei Shu, Liangchen Luo, Jayakumar Hoskere, Yun Zhu, Yinxiao Liu, Simon Tong, Jindong Chen, and Lei Meng. 2024. Rewritelm: An instruction-tuned large language model for text rewriting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 18970–18980.

Xiangru Tang, Tianyu Hu, Muyang Ye, Yanjun Shao, Xunjian Yin, Siru Ouyang, Wangchunshu Zhou, Pan Lu, Zhuosheng Zhang, Yilun Zhao, Arman Cohan, and Mark Gerstein. 2025. Chemagent: Self-updating memories in large language models improves chemical reasoning. In *The Thirteenth International Conference on Learning Representations*.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018. Fever: a large-scale dataset for fact extraction and verification. *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 809–819.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Rodriguez Aurelien, Joulin Armand, Grave Edouard, and Lample Guillaume. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022. Musique: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554.

Miles Turpin, Julian Michael, Ethan Perez, and Samuel Bowman. 2023. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. In *Advances in Neural Information Processing Systems*, volume 36, pages 74952–74965. Curran Associates, Inc.

Martin Tutek, Fateme Hashemi Chaleshtori, Ana Marasovic, and Yonatan Belinkov. 2025. Measuring chain of thought faithfulness by unlearning reasoning steps. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 9935–9960, Suzhou, China. Association for Computational Linguistics.

Guangya Wan, Yuqi Wu, Jie Chen, and Sheng Li. 2025. Reasoning aware self-consistency: Leveraging reasoning paths for efficient llm sampling. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 3613–3635. Association for Computational Linguistics.

Zilong Wang, Hao Zhang, Chun-Liang Li, Julian Martin Eisenschlos, Vincent Perot, Zifeng Wang, Lesly Miculicich, Yasuhisa Fujii, Jingbo Shang, Chen-Yu Lee, and Tomas Pfister. 2024. Chain-of-table: Evolving tables in the reasoning chain for table understanding. In *The Twelfth International Conference on Learning Representations*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.

Zhihui Xie, Jizhou Guo, Tong Yu, and Shuai Li. 2024. Calibrating reasoning in language models with internal consistency. In *Advances in Neural Information Processing Systems*, volume 37, pages 114872–114901. Curran Associates, Inc.

Wujiang Xu, Zujie Liang, Kai Mei, Hang Gao, Juntao Tan, and Yongfeng Zhang. 2025a. A-mem: Agentic memory for LLM agents. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*.

Yige Xu, Xu Guo, Zhiwei Zeng, and Chunyan Miao. 2025b. SoftCoT: Soft chain-of-thought for efficient reasoning with LLMs. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 23336–23351. Association for Computational Linguistics.

John Yang, Carlos E Jimenez, Alexander Wettig, Kilian Lieret, Shunyu Yao, Karthik Narasimhan, and Ofir Press. 2024. Swe-agent: Agent-computer interfaces enable automated software engineering. In *Advances in Neural Information Processing Systems*, volume 37, pages 50528–50652. Curran Associates, Inc.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 2369–2380.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik R Narasimhan, and Yuan Cao. 2023. React: Synergizing reasoning and acting in language models. In *The Eleventh International Conference on Learning Representations*.

Shaowei Zhang and Deyi Xiong. 2025. Debate4MATH: Multi-agent debate for fine-grained reasoning in math. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 16810–16824. Association for Computational Linguistics.

Tianhua Zhang, Jiaxin Ge, Hongyin Luo, Yung-Sung Chuang, Mingye Gao, Yuan Gong, Yoon Kim, Xixin Wu, Helen Meng, and James Glass. 2024. Natural language embedded programs for hybrid language symbolic reasoning. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 4131–4155. Association for Computational Linguistics.

Chengshuai Zhao, Zhen Tan, Pingchuan Ma, Dawei Li, Bohan Jiang, Yancheng Wang, Yingzhen Yang, and Huan Liu. 2025. Is chain-of-thought reasoning of llms a mirage? a data distribution lens. *arXiv preprint arXiv:2508.01191*.

Denny Zhou, Nathanael Schärli, Le Hou, Jason Wei, Nathan Scales, Xuezhi Wang, Dale Schuurmans, Claire Cui, Olivier Bousquet, Quoc Le, and Ed H. Chi. 2023. Least-to-most prompting enables complex reasoning in large language models. In *The Eleventh International Conference on Learning Representations*.

A Violation Types and Audit Schema

A.1 Violation Types

To enable structured and auditable reasoning verification, we formalize a taxonomy of reasoning faithfulness violations at the level of intermediate reasoning steps. Each violation type corresponds to a distinct way in which a reasoning step may lack sufficient logical or evidential support under the information available to the agent. This step-level formulation allows violations to be precisely localized and subsequently repaired without regenerating the entire trajectory (Turpin et al., 2023; Tutek et al., 2025). We consider the following violation types, which commonly arise in agentic reasoning settings:

- **Missing_Assumption**: A reasoning step implicitly relies on unstated assumptions that are necessary for the inference to hold. Such assumptions are neither established in prior steps nor supported by available evidence, making the step underspecified.
- **Invalid_Precondition**: The reasoning step depends on preconditions that have not been verified or do not hold under the current context, such as incorrect entity grounding or unmet factual constraints.
- **Unjustified_Inference**: The conclusion of the step is not sufficiently supported by preceding reasoning or evidence, often arising from heuristic extrapolation, pattern matching, or numerical guessing.
- **Circular_Reasoning**: The step directly or indirectly depends on its own conclusion, resulting in a logically invalid inference loop that provides no independent justification.
- **Contradiction**: The step conflicts with previously established reasoning steps or available evidence, indicating internal inconsistency within the trajectory.
- **Overgeneralization**: The step extrapolates beyond the scope of the supporting evidence, applying conclusions derived from specific cases to broader contexts without justification.

Each detected violation is represented as a tuple (t, l), where t denotes the violation type and l indexes the reasoning step at which the violation occurs. This structured representation serves as the interface between reasoning audit and subsequent repair, enabling targeted and minimal correction of unfaithful reasoning steps.

A.2 Audit Schema

As described in §3.4, the adversarial reasoning audit operates on selected belief trajectories to identify intermediate steps that violate reasoning faithfulness under the agents available information. To enable precise localization and subsequent repair, the audit produces a structured schema that serves as the explicit interface between reasoning verification and constraint-guided correction. Given a candidate reasoning trajectory $r = (s_1, \ldots, s_L)$, the auditor evaluates each step using the support function $\Gamma(\cdot)$ defined in §3.1. Steps for which $\Gamma(s_l \mid x, \mathcal{H}_l, \mathcal{E}_l) < \epsilon$ are flagged as unfaithful. The audit outcome is first represented as a set of violation instances $\mathcal{V}(r) = \{(t_j, l_j)\}_{j=1}^{m}$, where each tuple identifies a localized reasoning failure. Here, $t_j \in \mathcal{T}$ denotes the violation type defined in §A.1, and $l_j \in \{1, \ldots, L\}$ indexes the reasoning step at which the violation occurs.

To support verifiable and targeted repair, each violation instance (t_j, l_j) is further augmented by the audit schema with two additional components. First, the auditor provides a concise explanation that grounds the violation in a specific logical or evidential deficiency. Second, the auditor specifies an acceptance criterion θ_j, which defines a concrete and checkable condition under which the violation is considered resolved. Together, these components transform each violation instance into a repair-ready diagnostic unit.

The acceptance criterion θ_j plays a central role in operationalizing reasoning faithfulness. Rather than relying on free-form critique or subjective rewriting, the auditor translates each detected violation into an explicit constraint, such as requiring citation to retrieved evidence, clarification of an implicit assumption, or correction of an invalid inference. This design ensures that subsequent repair decisions are guided by verifiable conditions rather than stylistic preferences, and directly supports the constraint-guided repair objective described in §3.5.

By explicitly associating each violation with a step index, an explanation, and a corresponding acceptance criterion, the audit schema enables minimal counterfactual intervention: only the localized failing slices need to be edited, while unaffected steps remain unchanged. This step-level, constraint-oriented representation distinguishes our approach from surface-level critique or global rewriting methods, and provides the structural foundation for iterative audit–repair before belief commitment in agentic systems.

B Acceptance Criterion for Violation

To ensure that detected reasoning violations can be resolved in a verifiable and auditable manner, we introduce explicit *acceptance criteria* associated with each violation instance identified by the auditor. An acceptance criterion specifies a concrete condition under which a previously unfaithful reasoning step is considered sufficiently supported and thus eligible for commitment to agent actions or memory. As defined in §A.2, the auditor first outputs a set of violation instances $\mathcal{V}(r) = \{(t_j, l_j)\}_{j=1}^{m}$. For each violation instance (t_j, l_j), the audit schema further augments it with an explanation e_j and an acceptance criterion θ_j. The acceptance criterion θ_j serves as a repair target that operationalizes when the localized violation is considered resolved.

Formally, each acceptance criterion θ_j defines a checkable predicate over a (repaired) reasoning trajectory r: $\mathsf{Sat}(r, \theta_j) \in \{0, 1\}$, which evaluates whether the criterion associated with the violation instance (t_j, l_j) is satisfied. A violation is considered resolved if and only if $\mathsf{Sat}(r, \theta_j) = 1$. A repaired reasoning trajectory is deemed faithful only when all associated acceptance criteria are satisfied. Crucially, acceptance criteria are designed to be *operational and localized*. Rather than prescribing how a reasoning step should be rewritten, each criterion specifies what must be explicitly established for the step to be considered faithful. Typical acceptance criteria include, but are not limited to: (i) explicit citation or extraction of supporting evidence from retrieved documents or tool outputs; (ii) clarification and explicit statement of previously implicit assumptions; (iii) verification of required preconditions, such as entity identity or factual constraints; (iv) removal of logical dependencies that induce circularity or contradiction.

By separating violation detection from resolution conditions, acceptance criteria prevent subjective or surface-level repair decisions. They serve as a strict gate for belief commitment: repaired reasoning trajectories that fail to satisfy any associated acceptance criterion are rejected, regardless of their fluency or plausibility. This design ensures that reasoning faithfulness is enforced independently of final-task correctness or consensus among alternative reasoning traces. Moreover, acceptance criteria enable *minimal counterfactual intervention*. Since each criterion is associated with a specific violation instance and reasoning step, repair procedures can target only the localized failure slice without regenerating unaffected parts of the trajectory. This property is essential for preserving auditability and preventing unintended drift during iterative audit–repair.

C Prompt Used

C.1 Persona Generation Prompt

Persona-Based Belief Generation

You are an internal reasoning persona of an autonomous agent.

Your goal is to solve the given task by producing a belief $y = (c, r)$ consisting of: (1) a final claim c (an answer or a refusal-to-commit decision), and (2) an explicit step-by-step reasoning trajectory r. Your reasoning will be audited for faithfulness before any action or memory update.

Persona Instruction:
{persona_instruction}

Task:
{input_task}

Available Information:
{retrieved documents, tool outputs, observations, or constraints available to the agent}

Output Requirements:

- Produce a complete reasoning trajectory with clearly separated steps.
- For each step, make explicit whether it relies on prior reasoning or on external evidence.
- If a step depends on assumptions that are not directly supported, state them explicitly.
- If the task cannot be resolved with the available information, output **Do not commit** as the final claim and explain why.

Output Format:

- **Claim** c**:** {final answer or "Do not commit"}
- **Reasoning** r**:**
  - Step 1: ...
  - Step 2: ...
  - ...

Constraints:

- Follow the persona instruction strictly to induce a distinct reasoning bias.
- Do not omit intermediate steps or compress multiple inferences into one step.
- Do not correct or preemptively audit your own reasoning.

C.2 Auditor Prompt

Reasoning Auditor

You are an internal reasoning auditor of an autonomous agent.

Your task is to examine the given reasoning trajectory and identify any intermediate steps that violate reasoning faithfulness. You must not propose alternative answers or rewrite the reasoning. Your goal is to localize unfaithful reasoning steps and produce structured, verifiable diagnostics for subsequent repair.

Task:
{input_task}

Reasoning Trajectory to Audit:
{step-by-step reasoning trajectory}

Available Evidence:
{retrieved documents, tool outputs, or observations}

Audit Instructions:

- Evaluate each reasoning step independently and identify any step that lacks sufficient logical or evidential support.
- For each detected issue, localize the *failing slice* of the reasoning step.
- Formulate a concrete *probe* that checks whether the step satisfies its required support conditions.
- Based on the probe outcome, assign an appropriate violation type.
- Do not assume missing evidence or repair the reasoning.

Violation Types:

- `Missing_Assumption`: required assumptions are implicit or unstated.
- `Invalid_Precondition`: the step relies on conditions not established earlier.
- `Unjustified_Inference`: conclusions are drawn without sufficient support.
- `Circular_Reasoning`: the step depends on its own conclusion.
- `Contradiction`: the step conflicts with previous steps or available evidence.
- `Overgeneralization`: the step extrapolates beyond the supported scope.

Output Format: For each detected violation, output a structured audit record:

- **Step Index:** the index of the violated reasoning step.
- **Failing Slice:** the minimal fragment of the reasoning step that triggers the violation.
- **Probe:** a diagnostic question or check used to assess the step's faithfulness.
- **Violation Type:** one of the types listed above.
- **Acceptance Criterion:** a concrete, checkable condition that would resolve the violation.

If no violations are detected, output `No Violations Found`.

C.3 Repair Prompt

Faithful Reasoning Repair

You are an internal reasoning repair module of an autonomous agent.

Your task is to minimally repair a reasoning trajectory that has been flagged by an auditor as unfaithful. You must preserve all valid parts of the original reasoning and modify *only* the localized failing slices identified by the audit.

Inputs:

- **Original Task:**
  {input_task}
- **Original Belief:**
  A belief $y = (c, r)$ consisting of a claim and a step-by-step reasoning trajectory.
- **Audit Record(s):**
  One or more structured audit records, each specifying:
  - the violated step index,
  - the failing slice,
  - the violation type, and
  - the acceptance criterion.
- **Available Evidence:**
  {retrieved documents, tool outputs, or observations}

Repair Instructions:

- Apply *minimal edits* that are sufficient to satisfy each acceptance criterion.
- Do not modify reasoning steps that are not explicitly flagged by the audit.
- Do not introduce new assumptions unless required by the acceptance criterion.
- If new evidence extraction or computation is required, bind it explicitly to the available evidence.
- Ensure that the repaired reasoning remains internally consistent and task-relevant.

Output Requirements:

- Output a repaired belief $\tilde{y} = (\tilde{c}, \tilde{r})$.
- Clearly indicate which reasoning steps were edited, added, or deleted.
- The final claim $\tilde{c}$ must follow logically from the repaired reasoning trajectory.

Constraints:

- Do not re-audit the reasoning.
- Do not optimize, refactor, or simplify unaffected parts of the reasoning.
- Do not introduce stylistic or heuristic improvements beyond what is required for faithfulness.

D Detailed Dataset Descriptions

We evaluate SAVER on six benchmarks spanning multi-hop reasoning, evidence-sensitive decision making, and local reasoning. Table 4 summarizes the dataset sizes for our evaluation benchmarks.

HotpotQA. HotpotQA is a multi-hop question answering benchmark designed to require reasoning over multiple Wikipedia articles. It provides question–answer pairs together with supporting facts (sentence-level supervision) that indicate which sentences are necessary for answering the question. The dataset includes both bridge-type questions (where one entity leads to another) and comparison-type questions (requiring aggregations such as greater/less, earlier/later). These properties make HotpotQA particularly suitable for faithfulness evaluation: a model can produce a correct final answer while still relying on unsupported intermediate claims or skipping required evidence hops.

2WikiMHQA. 2WikiMHQA extends multi-hop QA to a larger set of compositional questions constructed from Wikipedia. Similar to HotpotQA, it emphasizes multi-step reasoning (e.g., entity bridging, set intersection, and comparisons) across multiple documents. Compared to single-context QA, the reasoning process is more fragile: incorrect or unverified intermediate bindings (e.g., the wrong entity or attribute) can silently propagate, yielding coherent but unfaithful trajectories. We use this benchmark to test whether SAVER can localize and repair hop-level violations before commitment.

Dataset	Setting	Train	Dev	Test	Total
HotpotQA	Multi-hop QA	90,447	7,405	7,405	105,257
2WikiMHQA	Multi-hop QA	167,454	12,576	12,576	192,606
MuSiQue	Multi-hop QA	19,938	2,417	2,459	24,814
NQ	Evidence-sensitive QA	307,373	7,830	7,842	323,045
FEVER	Evidence-sensitive QA	145,449	19,998	19,998	185,445
Quoref	Local reasoning	19,399	2,418	2,537	24,354

Table 4: Dataset sizes for our evaluation benchmarks.

MuSiQue. MuSiQue is a multi-hop QA benchmark explicitly constructed to encourage *composi-*

tional reasoning rather than dataset artifacts. Questions are organized around multi-step chains with intermediate sub-questions, and instances often exhibit entity ambiguity and attribute-binding risk (e.g., a year associated with a publication vs. a parent institution). These characteristics make MuSiQue a strong stress test for reasoning faithfulness, as agents must maintain correct hop dependencies and avoid entity/attribute slippage under multi-step processing.

Natural Questions (NQ). Natural Questions is a large-scale open-domain QA dataset derived from real Google search queries. Questions vary widely in type and frequently require disambiguation, scope control, and evidence-grounded naming (e.g., formal titles or role names). Because the questions are naturalistic and often underspecified, models may introduce implicit assumptions (jurisdiction, version, or referent) that are not warranted by available evidence. We include NQ to evaluate whether SAVER can prevent unsupported assumptions from being committed into the agent's belief state and downstream decisions.

FEVER. FEVER is a fact verification benchmark where the task is to classify a claim as SUPPORTED, REFUTED, or NOTENOUGHINFO given evidence from Wikipedia. The key challenge is *evidence sufficiency*: a model can generate plausible rationales while lacking the necessary evidence, and the correct label may depend on whether the evidence fully entails or contradicts the claim. This aligns closely with our definition of step-level faithfulness, where intermediate reasoning steps must be justified by accessible evidence before commitment.

Quoref. Quoref is a reading comprehension benchmark focusing on coreference resolution and referential reasoning within a single passage. Compared to open-domain and multi-hop settings, Quoref reduces structural uncertainty from retrieval and multi-document integration, providing a controlled setting where failures are more directly attributable to local reasoning errors. We use Quoref as a baseline to examine whether SAVER still provides benefits when evidence is already contained in one context and the primary difficulty lies in resolving referential dependencies.

Overall, these benchmarks cover complementary reasoning regimes: (i) *multi-hop QA* (HotpotQA, 2WikiMHQA, MuSiQue) tests hop dependency tracking and cross-document evidence binding; (ii) *evidence-sensitive QA* (NQ, FEVER) stresses assumption control and evidence sufficiency; and (iii) *local reasoning* (Quoref) isolates faithfulness under minimal retrieval-induced uncertainty. This diversity allows us to evaluate SAVER across settings where unfaithful intermediate beliefs may arise from different structural sources.

E Faithfulness Metrics Calculation

We provide the precise trajectory-level definitions of all faithfulness metrics used in our evaluation. Importantly, all metrics are computed *per trajectory* and then averaged across trajectories (i.e., macro aggregation), rather than aggregated globally over all reasoning steps. This design avoids over-weighting longer trajectories.

For a trajectory r_i with L_i reasoning steps, the auditor produces a set of detected violation instances $V(r_i) = \{(t_{i,j}, \ell_{i,j})\}_{j=1}^{m_i}$, where $t_{i,j}$ denotes the violation type, $\ell_{i,j} \in \{1, \dots, L_i\}$ denotes the triggered step index, and $m_i = |V(r_i)|$ is the total number of detected violations for trajectory r_i. A reasoning step is considered unfaithful if it is associated with at least one violation instance in $V(r_i)$. Based on these auditor outputs, we define the following metrics over a dataset of N trajectories:

- **Average Violations (Avg Viol).** This metric measures the average number of detected faithfulness violations per trajectory: Avg Viol $= \frac{1}{N}\sum_{i=1}^{N} |V(r_i)|$.
- **Violation-Free Rate (VFR).** This metric measures the proportion of trajectories that contain no detected violations: VFR $= \frac{1}{N}\sum_{i=1}^{N} \mathbb{I}(|V(r_i)| = 0)$, where $\mathbb{I}(\cdot)$ is the indicator function.
- **Post-Repair Residual (Post-Res).** To evaluate the residual unfaithfulness after repair, we re-run the auditor on the repaired trajectory $\tilde{r}_i$ and compute the average number of remaining violations: Post-Res $= \frac{1}{N}\sum_{i=1}^{N} |V(\tilde{r}_i)|$.
- **Unfaithful Step Rate (USR).** This metric measures the average proportion of reasoning steps in a trajectory that are flagged as unfaithful. Since multiple violation instances may be attached to the same step, each step is counted at most once: USR $= \frac{1}{N}\sum_{i=1}^{N} \frac{1}{L_i}\sum_{\ell=1}^{L_i} \mathbb{I}(\exists j \text{ such that } \ell_{i,j} = \ell)$.

Methods	HotpotQA				2WikiMHQA				MuSiQue			
	Avg Viol ↓	VFR ↑	Post-Res ↓	USR ↓	Avg Viol ↓	VFR ↑	Post-Res ↓	USR ↓	Avg Viol ↓	VFR ↑	Post-Res ↓	USR ↓
VANILLA LM	2.40	9.23%	–	42.84%	2.55	8.10%	–	49.66%	2.95	6.48%	–	58.25%
COT	1.75	27.33%	–	24.92%	1.95	19.87%	–	29.46%	2.65	15.12%	–	33.67%
MAD	1.15	39.67%	–	21.85%	1.63	35.45%	–	25.97%	1.95	29.74%	–	30.45%
SAVER(Ours)	0.31	84.20%	0.03	7.78%	0.49	75.64%	0.06	11.20%	0.71	72.28%	0.09	15.60%

Table 5: Reasoning Faithfulness Evaluation on Multi-hop QA Benchmarks under Qwen-2.5-7B.

Methods	LLaMA-3.1-8B							Qwen-2.5-7B						
	Avg Tokens (k)	Avg Time (s)	API Calls	Avg Viol ↓	VFR ↑	Post-Res ↓	USR ↓	Avg Tokens (k)	Avg Time (s)	API Calls	Avg Viol ↓	VFR ↑	Post-Res ↓	USR ↓
VANILLA LM	2.3	2.2	1	2.65	7.43%	–	46.41%	2.5	2.2	1	2.40	9.23%	–	42.84%
COT	2.8	3.1	1	1.98	24.89%	–	27.36%	3.1	3.3	1	1.75	27.33%	–	24.92%
MAD	11.2	11.5	6	1.33	36.74%	–	23.94%	11.9	11.6	6	1.15	39.67%	–	21.85%
SELF-REFINE	7.6	8.7	4	1.48	31.80%	0.18	21.77%	8.1	8.9	4	1.24	36.13%	0.23	19.94%
B-2	4.1	4.8	2	1.62	28.57%	–	23.61%	4.3	4.8	2	1.46	32.83%	–	20.85%
SAVER(Ours)	12.4	13.4	8	0.37	81.36%	0.05	9.12%	12.8	12.4	8	0.31	84.20%	0.03	7.78%

Table 6: The computational cost comparison of SAVER and other baseline methods on HotpotQA.

F Additional Experimental Results

F.1 Faithfulness Evaluation

Table 5 reports the reasoning faithfulness evaluation on three multi-hop QA benchmarks using Qwen-2.5-7B as the backbone model. We observe that SAVER exhibits consistent improvements over all baselines across datasets, achieving lower average violation counts and unsupported step rates, together with substantially higher violation-free reasoning ratios. While CoT and MAD reduce unfaithful reasoning to some extent, they continue to produce a non-negligible number of unsupported intermediate steps. In contrast, SAVER maintains low post-repair residual violations, suggesting that the auditrepair mechanism remains effective under a different model backbone, which further confirms that the gains of SAVER are robust across backbones.

F.2 Computational Overhead Analysis

We analyze computational overhead on HotpotQA using three efficiency indicators: average token consumption, end-to-end wall-clock latency, and the number of API calls per query. As shown in Table 6, SAVER is more expensive than single-pass prompting methods because it performs iterative auditing and repair. Under LLaMA-3.1-8B, SAVER requires 12.4k tokens, 13.4s latency, and 8 API calls per query, compared with 2.3k–2.8k tokens, 2.2–3.1s, and one call for Vanilla LM and CoT. A similar pattern holds for Qwen-2.5-7B, where SAVER uses 12.8k tokens, 12.4s latency, and 8 API calls. Despite this overhead, SAVER achieves the best faithfulness performance among all methods. On LLaMA-3.1-8B, it reduces Avg Viol to 0.37 and USR to 9.12%, while increasing VFR to 81.36%; on Qwen-2.5-7B, the corresponding results are 0.31, 7.78%, and 84.20%. Compared with other multi-call baselines such as MAD and Self-Refine, SAVER delivers substantially larger faithfulness gains at a comparable level of multi-call inference cost, indicating a favorable trade-off between computation and reasoning reliability.

F.3 Results on Non-QA Tasks

To assess generalization beyond QA, we evaluate SAVER on math reasoning and code generation. As shown in Table 7, SAVER achieves the best end-task performance on all four benchmarks, reaching 88.4/58.2 EM on GSM8K/MATH and 76.5/76.1 Pass@1 on HumanEval/MBPP. These results show that the gains of SAVER transfer beyond QA to numerical reasoning and program synthesis. The faithfulness results show the same trend. On GSM8K, SAVER achieves 0.24 Avg Viol, 83.44% VFR, and 6.53% USR; on HumanEval, it further improves to 0.12 Avg Viol, 88.46% VFR, 0.02 Post-Res, and 4.88% USR. Compared with strong multi-call baselines, SAVER consistently yields fewer violations and more violation-free trajectories, indicating that its audit–repair mechanism improves both final task performance and intermediate reasoning faithfulness across domains.

F.4 Ablation Study

Table 8 reports the ablation results under the Qwen-2.5-7B backbone on HotpotQA and

Methods	GSM8K	MATH	HumanEval	MBPP	GSM8K				HumanEval			
	EM ↑	EM ↑	Pass@1 ↑	Pass@1 ↑	Avg Viol ↓	VFR ↑	Post-Res ↓	USR ↓	Avg Viol ↓	VFR ↑	Post-Res ↓	USR ↓
VANILLA LM	79.2	44.3	66.8	68.7	1.58	18.77%	–	28.74%	1.37	22.75%	–	27.12%
COT	84.5	51.9	69.2	70.3	1.19	32.86%	–	23.95%	1.12	35.82%	–	19.03%
MAD	87.6	55.8	73.5	74.6	0.83	43.26%	–	19.31%	0.84	48.96%	–	15.76%
SELF-REFINE	85.3	53.4	72.4	73.8	0.98	38.94%	0.13	18.62%	0.77	52.79%	0.15	14.84%
B-2	86.8	54.8	73.9	74.2	0.92	41.27%	–	19.11%	0.82	46.38%	–	16.83%
SAVER(Ours)	**88.4**	**58.2**	**76.5**	**76.1**	**0.24**	**83.44%**	**0.03**	**6.53%**	**0.12**	**88.46%**	**0.02**	**4.88%**

Table 7: The end-task performance of SAVER and other baseline methods on *Math Reasoning* and *Code Generation* tasks. The best-performed method is marked by **bold**.

Methods	HotpotQA						2WikiMHQA					
	EM ↑	F1 ↑	Avg Viol ↓	VFR ↑	Post-Res ↓	USR ↓	EM ↑	F1 ↑	Avg Viol ↓	VFR ↑	Post-Res ↓	USR ↓
w/o Persona	43.6	52.7	0.44	76.83%	0.05	10.93%	47.9	55.7	0.71	68.14%	0.08	17.37%
w/o k-DPP	43.5	52.6	0.56	73.63%	0.07	14.29%	47.7	55.5	0.77	63.34%	0.12	19.62%
w/o Auditing	44.0	53.2	1.2	45.20%	–	24.57%	48.2	56.3	1.58	41.25%	–	27.44%
w/o Repair	44.3	53.2	1.38	35.88%	–	35.09%	48.4	56.2	1.65	31.24%	–	37.36%
SAVER	43.9	53.0	0.32	84.71%	0.03	7.54%	48.1	55.9	0.49	76.27%	0.06	11.20%

Table 8: Ablation Study on the HotpotQA and 2WikiMHQA Benchmarks under Qwen-2.5-7B.

2WikiMHQA. Overall, we observe trends consistent with those reported in the main paper. Removing individual components has a limited impact on EM and F1, but consistently worsens faithfulness-related metrics, indicating that the proposed modules primarily target intermediate reasoning quality rather than surface-level accuracy. In particular, disabling persona-conditioned belief generation increases both Avg Viol and USR, suggesting reduced coverage over diverse reasoning patterns. Eliminating structure-aware belief selection further amplifies unfaithful steps, reflecting the risk of correlated reasoning errors when diversity is not explicitly controlled. The most pronounced degradation occurs when either the auditing or repair module is removed, leading to substantially higher violation rates. These results further corroborate that adversarial auditing and constraint-guided repair are critical for enforcing reasoning faithfulness across different backbone models.

G Additional Case Study

To complement the main experiments, we present three case studies that illustrate how SAVER enforces reasoning faithfulness *prior* to action or memory commitment.

Case study in 2WikiMHQA stresses multi-hop numerical reasoning with explicit comparisons. We show two common unfaithful patterns: (i) heuristic conclusions derived from superficial cues, e.g., era-based generalization; (ii) locally inconsistent arithmetic that contradicts the extracted dates. SAVER audits the reasoning at the step level, pinpoints the failing slices, and performs minimal repairs that bind the comparison to evidence-grounded birth/death dates and corrected computations.

Case study in MuSiQue highlights multi-hop entity binding under ambiguity. The reasoning can appear evidence-based while silently switching the target entity, e.g., using the newspaper's founding year to answer an institute's founding-year query, or mapping the name "The Collegian" to an unsupported owner. SAVER localizes the precise step where the entity/attribute binding breaks, then repairs only that slice by enforcing an explicit ownership binding and extracting the institute's founding year from the correct evidence context.

Case study in NQ demonstrates failures that look linguistically well-formed but are not faithful, including circular justification via paraphrase and missing contextual assumptions. SAVER identifies the circular slice, e.g., treating "allegiance" as evidence for the formal name, and the implicit jurisdiction assumption, then repairs them by adding evidence-bound naming steps and explicitly stating and supporting the required context before commitment.

These three cases cover complementary faithfulness failures, and SAVER goes beyond surface rewriting by producing auditable diagnostics, applying verifiable minimal repairs, and committing beliefs only after all acceptance criteria are satisfied across all settings.

Case study in 2WikiMHQA

Input
Q: Who lived longer, Giuseppe Cesari or Nicos Poulantzas?

Belief Generation
Persona coalition $\mathcal{A} = \{a_1, a_2, a_3, a_4\}$ outputs beliefs $y_i = (c_i, r_i)$:

- y_1 **(evidence-first / protocol persona)**
  Claim c_1**:** Do not commit an answer until both birth/death dates are extracted and lifespans are computed.
  Reasoning r_1**:**

  1. Retrieve evidence passages for *Giuseppe Cesari* and *Nicos Poulantzas*.
  2. Extract each person's birth date and death date as exact spans from evidence.
  3. Compute each lifespan from the extracted dates and compare the two values before answering.

- y_2 **(template / intuition persona)**
  Claim c_2**:** Giuseppe Cesari
  Reasoning r_2**:**

  1. Giuseppe Cesari is a much earlier historical figure than Nicos Poulantzas.
  2. Earlier historical figures (from pre-modern eras) generally lived longer than modern political theorists.
  3. Therefore, Giuseppe Cesari lived longer.

- y_3 **(calculation / comparison persona)**
  Claim c_3**:** Nicos Poulantzas
  Reasoning r_3**:**

  1. From evidence: Giuseppe Cesari (1568–1640); Nicos Poulantzas (1936–1979).
  2. Compute lifespans: Cesari lived $1640 - 1568 = 72$ years; Poulantzas lived $1979 - 1936 = 73$ years.
  3. Since $73 > 72$, Nicos Poulantzas lived longer.

- y_4 **(audit persona)**
  Claim c_4**:** Reject commitment unless the chain (dates $\rightarrow$ lifespan $\rightarrow$ comparison) is evidence-bound and internally consistent.
  Reasoning r_4**:**

  1. Identify commitment requirements: (i) both persons' birth/death dates must be citable from evidence; (ii) lifespan arithmetic must match those dates; (iii) the comparison conclusion must follow from the computed values.
  2. Check y_2: if it uses era-based stereotypes instead of extracting dates and computing lifespans, flag overgeneralization.
  3. Check y_3: verify each subtraction against the cited dates; if any computed lifespan conflicts with the dates (or contains arithmetic errors), flag contradiction.
  4. If any requirement fails, do not commit; request repair by binding to evidence and editing only the minimal failing slice.

Belief Selection
After k-DPP sampling, select an audit subset $S = \{y_2, y_3\}$:

- Select y_2 to cover the high-risk heuristic pattern (era-based generalization).

- Select y_3 to cover the computation-based path (susceptible to arithmetic inconsistencies).

Audit
Audit y_2:

- **Failing slice:**
  "Earlier historical figures generally lived longer $\rightarrow$ therefore Giuseppe Cesari lived longer."
- **Probe:** Does the trace extract both birth/death dates and compute lifespans, or does it generalize from era stereotypes?
- **Output:** `Overgeneralization`

Audit y_3:

- **Failing slice:**
  "Poulantzas lived $1979 - 1936 = 73$ years."
- **Probe:** Is each computed lifespan numerically consistent with the cited birth/death dates?
- **Output:** `Contradiction`

Repair
Repair for y_2 (`Overgeneralization`):

- **R1:** Delete the era-based generalization ("earlier figures generally lived longer").
- **R2:** Add evidence extraction steps for both people: cite the birth date and death date spans for Cesari and Poulantzas.
- **R3:** Compute lifespans from the extracted dates and compare them before answering (derive the conclusion from the computed values).
- **Acceptance criteria:** The repaired belief must (i) cite birth/death date spans for each person, and (ii) show a lifespan-based comparison and its derived conclusion.

Repair for y_3 (`Contradiction`):

- **R1:** Keep the extracted dates unchanged; edit only the incorrect arithmetic slice.
- **R2:** Recompute Poulantzas' lifespan from the cited dates: $1979 - 1936 = 43$ (not 73).
- **R3:** Update the comparison conclusion accordingly: $72 > 43 \Rightarrow$ Giuseppe Cesari lived longer.
- **Acceptance criteria:** Every lifespan computation must match the cited dates, and the final comparison must follow those corrected values.

Re-audit after Repair (Verified Commit)

- Re-audit $\tilde{y}_2$: generalization removed; dates extracted; lifespans computed and compared $\rightarrow$ pass.
- Re-audit $\tilde{y}_3$: arithmetic consistent with dates; comparison updated $\rightarrow$ pass.

Commit: Answer = Giuseppe Cesari. ✓
Safe memory write: store only (Giuseppe Cesari: birth/death, lifespan; Nicos Poulantzas: birth/death, lifespan; comparison result), not era-based heuristics or inconsistent arithmetic.

Case study in MuSiQue

Input
Q (MuSiQue): When was the institute that owned *The Collegian* founded?

Belief Generation
Persona coalition $\mathcal{A} = \{a_1, a_2, a_3, a_4\}$ outputs beliefs $y_i = (c_i, r_i)$:

- y_1 **(name-matching / shortcut persona)**
  Claim c_1**:** 1844
  Reasoning r_1**:**

  1. "The Collegian" sounds like a well-known college newspaper; I associate it with Hillsdale College.
  2. Hillsdale College was founded in 1844.
  3. Therefore, the institute that owned *The Collegian* was founded in 1844.

- y_2 **(evidence-first but entity-slip persona)**
  Claim c_2**:** 1963
  Reasoning r_2**:**

  1. Find the passage that states who *The Collegian* is owned by (the "institute").
  2. In the same evidence block, locate a founding year.
  3. The passage says *The Collegian* "was founded in 1963" (as a newsletter).
  4. Therefore, the institute that owned *The Collegian* was founded in 1963.

- y_3 **(disambiguation persona)**
  Claim c_3**:** Do not answer until the exact *The Collegian* instance is pinned down by ownership evidence.
  Reasoning r_3**:**

  1. "The Collegian" is ambiguous: multiple schools have publications with that name.
  2. First extract the owner institute from an explicit sentence (e.g., "*The Collegian* is the ... publication of X").
  3. Then extract the founding year of that institute (not the newspaper) from evidence.

- y_4 **(audit persona)**
  Claim c_4**:** Reject commitment unless (owner is disambiguated) AND (founding year is for the owner institute).
  Reasoning r_4**:**

  1. Define commitment requirements:
     (i) A citable sentence must identify which *The Collegian* this is by stating its owner institute.
     (ii) A citable sentence must state the founding year of that institute (not the newspaper).
  2. Check y_1: if it jumps from the name "The Collegian" to a specific school without evidence, flag a missing assumption.
  3. Check y_2: if it uses the newspaper's founding year to answer an institute founding-year question, flag invalid precondition (entity/attribute mismatch).
  4. If any requirement fails, do not commit; request repair to (a) disambiguate owner, and (b) bind the year to the owner institute.

Belief Selection
After k-DPP sampling, select an audit subset $S = \{y_2, y_1\}$:

- Select y_2 to cover the *evidence-present but wrong-entity* failure pattern (high risk in multi-hop QA).
- Select y_1 to cover the *name-matching shortcut* pattern (high ambiguity risk).

Audit
Audit y_2:

- **Failing slice:**
"*The Collegian* was founded in 1963 $\rightarrow$ the institute that owned it was founded in 1963."
- **Probe:** Does the cited year describe the **owner institute** (university), or the **newspaper/publication**?
- **Output:** `Invalid_Precondition`

Audit y_1:

- **Failing slice:**
"'*The Collegian*' refers to Hillsdale College $\rightarrow$ answer using Hillsdale's founding year."
- **Probe:** Is the mapping from the string name "The Collegian" to a specific institute explicitly supported by an ownership sentence?
- **Output:** `Missing_Assumption`

Repair
Repair for y_2 (`Invalid_Precondition`):

- **R1:** Remove the step that treats the newspaper's founding year as the institute's founding year.
- **R2:** Add an ownership binding step: extract the institute that owns/publishes *The Collegian* from an explicit sentence.
- **R3:** Add an institute-year extraction step: extract the founding year of that institute (e.g., "Houston Baptist University ... was founded in 1960").
- **Acceptance criteria:** The final year must be supported by a sentence whose subject is the **owner institute**, not the newspaper.

Repair for y_1 (`Missing_Assumption`):

- **R1:** Insert an explicit disambiguation step: "Identify which *The Collegian* by extracting its owner institute from evidence."
- **R2:** Replace the Hillsdale association with the evidence-identified owner institute (e.g., Houston Baptist University).
- **R3:** Under that evidence-grounded owner, extract the institute's founding year from evidence (1960).
- **Acceptance criteria:** The repaired belief must include a citable ownership sentence that resolves the ambiguity, before using any founding year.

Re-audit after Repair (Verified Commit)

- Re-audit $\tilde{y}_2$: year is now attached to the owner institute $\rightarrow$ pass.
- Re-audit $\tilde{y}_1$: ambiguity resolved by ownership evidence; no unstated mapping $\rightarrow$ pass.

Commit: Answer = 1960.✓
Safe memory write: store only (owner institute identified from evidence $\rightarrow$ institute founded in 1960), not the name-matching shortcut or the newspaper-year confusion.

Case study in NQ

Input
Q: What is the oath that new citizens take?

Belief Generation
Persona coalition $\mathcal{A} = \{a_1, a_2, a_3, a_4\}$ outputs beliefs $y_i = (c_i, r_i)$:

- y_1 **(template / intuition persona)**
  Claim c_1: Oath of Allegiance
  Reasoning r_1:
  1. New citizens take an oath of "allegiance."
  2. Therefore, the oath is called the Oath of Allegiance.
- y_2 **(evidence-first persona)**
  Claim c_2: Do not answer until the formal name is extracted from evidence.
  Reasoning r_2:
  1. Locate an evidence passage about "new citizens/naturalization oath."
  2. Extract the formal oath name as an exact span from the same sentence/paragraph.
- y_3 **(disambiguation persona)**
  Claim c_3: Likely the U.S. Oath of Allegiance, but the country context must be stated and supported.
  Reasoning r_3:
  1. "New citizens" can refer to different countries' naturalization processes.
  2. The country/jurisdiction must be made explicit; otherwise, the answer is underspecified.
  3. If evidence indicates U.S. naturalization, extract the formal oath name span.
- y_4 **(audit persona)**
  Claim c_4: Reject commitment unless both (context + formal-name evidence) are satisfied.
  Reasoning r_4:
  1. Identify commitment requirements: (i) jurisdiction/context must be explicit; (ii) the oath's formal name must appear in a citable evidence sentence.
  2. Check y_1: if it only paraphrases "allegiance" into "Oath of Allegiance" without a naming sentence, flag circular support.
  3. Check y_3: if it assumes "U.S." without stating or citing the context, flag missing assumption.
  4. If any requirement fails, do not commit; instead request repair by adding the missing assumption and/or the naming evidence sentence.

Belief Selection
After k-DPP sampling, select an audit subset $S = \{y_1, y_3\}$:

- Select y_1 to cover the high-risk paraphrase→answer pattern (often circular).
- Select y_3 to cover the context-sensitive path where assumptions can be implicit.

Audit
Audit y_1:

- **Failing slice:**
  "New citizens swear allegiance → therefore the oath is called Oath of Allegiance."
- **Probe:** Does the trace cite a sentence that formally names the oath, or is it just restating the conclusion?
- **Output:** `Circular_Reasoning`

Audit y_3:

- **Failing slice:**
  "Assume 'new citizens' means U.S. naturalization → answer with the U.S. oath name." (if not explicitly stated/supported)
- **Probe:** Is the jurisdiction explicitly stated and supported by evidence before committing the name?
- **Output:** `Missing_Assumption`

Repair
Repair for y_1 (`Circular_Reasoning`):

- **R1:** Delete the paraphrase-based justification ("allegiance → Oath of Allegiance").
- **R2:** Add an evidence lookup step: find a passage that explicitly names the naturalization oath.
- **R3:** Extract the formal name span from an evidence sentence, e.g.,
  "The naturalization oath is formally called the United States Oath of Allegiance."
- **Acceptance criteria:** A citable evidence sentence must explicitly bind "the naturalization oath/oath new citizens take" to its formal name span.

Repair for y_3 (`Missing_Assumption`):

- **R1:** Add an explicit assumption statement:
  "Here 'new citizens' refers to U.S. naturalization (as indicated by the evidence context)."
- **R2:** Attach context evidence (e.g., mentions of "United States / USCIS / naturalization").
- **R3:** Under that stated context, extract the same formal oath name span from evidence.
- **Acceptance criteria:** The repaired belief must contain both (i) an explicit jurisdiction assumption, and (ii) a citable naming sentence for the oath.

Re-audit after Repair (Verified Commit)

- Re-audit $\tilde{y}_1$: circular slice removed; formal-name evidence bound → pass.
- Re-audit $\tilde{y}_3$: missing assumption explicit + supported; name extracted from evidence → pass.

Commit: Answer = United States Oath of Allegiance.✓
Safe memory write: store only (jurisdiction: U.S. naturalization → formal oath name = United States Oath of Allegiance), not the paraphrase template or implicit assumptions.