LogicVault: Persistent Symbolic Belief States for Cross-Query Logical Consistency in LLMs

Sarim Chaudhry
Purdue University
chaud158@purdue.edu

Abstract

Large Language Models (LLMs) answer each query in isolation with no persistent logical state. This causes contradictions across related questions. We introduce LogicVault, a framework that maintains a symbolic belief vault alongside any LLM and enforces cross-query consistency through an external SMT solver. For each response, LogicVault formalizes the output into first-order logic, checks it against all prior beliefs via Z3, and repairs contradictions by feeding the minimal unsatisfiable core back to the LLM. A belief revision module based on AGM theory handles genuine world-model updates. We release LogicBench-Cross, the first benchmark for cross-query logical consistency, containing 500 multi-query scenarios across five domains. Across six LLMs, LogicVault reduces cross-query contradictions by 78% and improves single-query accuracy on FOLIO, ProofWriter, and LogiQA 2.0. The framework requires no training and works with any LLM at inference time. Code is available at: https://github.com/Sarimsaljook/LogicVault.

1 Introduction

Large Language Models (LLMs) frequently generate contradictory responses across related contexts, even when possessing the requisite domain knowledge. For example, a model might correctly state that NSAIDs are contraindicated for patients with gastric ulcers, while separately advising that Ibuprofen is safe for the same condition. This inconsistency stems from the structural independence of standard inference: because queries are processed in isolation, no intrinsic mechanism ensures that the response to a current query q_n remains logically compatible with the assertions made in responses to prior queries $q_1, \ldots, q_{n-1}$.

The problem is well-documented. Ghosh et al. (2025) show that LLaMA-2-70B answers true to both "Is an albatross an organism?" and "Is an albatross *not* an organism?" Mündler et al. (2024) find pervasive self-contradiction within single outputs. Calanzone et al. (2025) establish that training on QA data alone does not produce consistency.

Existing approaches address fragments of this problem. BeliefBank (Kassner et al., 2021) stores belief-answer pairs with constraint propagation but uses no formal solver. Logic-LM (Pan et al., 2023) and Aristotle (Chen et al., 2025) invoke solvers per query but maintain no cross-query state. LoCo-LMs (Calanzone et al., 2025) and REPAIR (Liu et al., 2025) modify training losses but provide no inference-time guarantees.

We introduce **LogicVault**. A novel framework that maintains a persistent symbolic belief state verified by an external SMT solver. Each LLM response is formalized into first-order logic and checked against the vault for satisfiability. Contradictions trigger targeted repair using the minimal unsatisfiable core. Genuine world-model updates are handled through AGM belief revision (Alchourrón et al., 1985).

LogicVault requires no training, operates at inference time, and works with any LLM. We contribute:

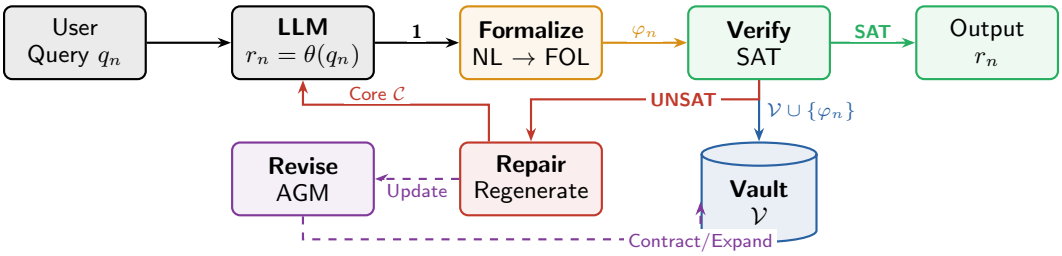


Figure 1: The LogicVault pipeline. Each LLM response is formalized into FOL, verified against the vault via Z3. SAT responses are accepted and added to the vault. UNSAT responses trigger repair with the minimal unsatisfiable core. Dashed arrows show the AGM belief revision path for genuine world-model updates.

1. The LogicVault framework that includes a persistent belief state, solver-in-the-loop verification, targeted repair, and AGM-based revision.
2. LogicBench-Cross: the first benchmark for cross-query logical consistency (500 scenarios, 5 domains, 3 difficulty levels).
3. Experiments on six LLMs showing 78% contradiction reduction and 9–13% accuracy gains without training.

2 Related Work

Logical consistency of LLMs. Kassner et al. (2021) introduced BeliefBank, which stores belief-answer pairs and applies constraint propagation. It handles simple negation but lacks formal solvers. Calanzone et al. (2025) propose LoCo-LMs, a neuro-symbolic training loss for consistency. Liu et al. (2025) introduce REPAIR for quantifying compositional consistency via transitivity, commutativity, and negation invariance. Both require training and provide no inference-time guarantees. Mündler et al. (2024) detect self-contradictions in single outputs reactively. Ghosh et al. (2025) provide systematic analysis of logical inconsistencies in fact-checking. All of these methods are either training-time, post-hoc, or limited to pairwise checks. LogicVault is proactive, inference-time, and cross-query.

Neuro-symbolic reasoning. Logic-LM (Pan et al., 2023) translates natural language to FOL and invokes Prover9/Z3 per query. LINC (Olausson et al., 2023) uses FOL provers for neurosymbolic reasoning. Aristotle (Chen et al., 2025) employs proof-by-contradiction with decompose-search-resolve. All operate on single queries. None maintain cross-query state.

External solver integration. Recent work shows adaptive symbolic language selection yields 96% accuracy on composite reasoning benchmarks (Wang et al., 2025). Solvers are invoked per-query and discarded. No system uses them for persistent belief maintenance.

Belief revision. AGM theory (Alchourrón et al., 1985) provides axiomatic foundations for rational belief change through expansion, contraction, and revision. Truth Maintenance Systems (Doyle, 1979) track belief dependencies. These classical AI ideas have not been applied to LLM consistency.

3 Methods

We define LogicVault as a system $\mathcal{L} = (\text{LLM}, \mathcal{V}, \mathcal{S}, \mathcal{R})$ where LLM is any language model (black-box), $\mathcal{V} = \{\varphi_1, \varphi_2, \ldots, \varphi_n\}$ is the persistent belief vault of first-order logic (FOL) propositions, $\mathcal{S}$ is an SMT solver (Z3; de Moura & Bjørner, 2008), and $\mathcal{R}$ is a belief revision module implementing AGM postulates. Figure 1 shows the pipeline.

For each query q_n, the pipeline executes four stages. **Stage 1 (Generate):** The LLM produces response r_n. **Stage 2 (Formalize):** A two-stage process extracts atomic claims from r_n and translates them into FOL propositions φ_n. A self-verification loop back-translates the FOL to natural language and checks semantic equivalence. This catches 85% of translation

Algorithm 1 LogicVault Inference

Require: Query q_n, Vault $\mathcal{V}$, Solver $\mathcal{S}$, LLM θ, max repairs k
Ensure: Consistent response r_n, updated vault $\mathcal{V}'$

$r_n \leftarrow \theta(q_n)$
for $i = 1$ to k **do**
  $\varphi_n \leftarrow \textsc{Formalize}(r_n)$; $\varphi_n \leftarrow \textsc{SelfVerify}(\varphi_n, r_n)$
  if $\mathcal{S}.\textsc{Sat}(\mathcal{V} \cup \{\varphi_n\})$ **then**
    return r_n, $\mathcal{V} \cup \{\varphi_n\}$
  else
    $\mathcal{C} \leftarrow \mathcal{S}.\textsc{UnsatCore}(\mathcal{V} \cup \{\varphi_n\})$
    if $\textsc{IsWorldUpdate}(q_n, r_n, \mathcal{C})$ **then**
      return r_n, AGM-$\textsc{Revise}(\mathcal{V}, \varphi_n)$
    end if
    $r_n \leftarrow \theta(q_n, \textsc{Explain}(\mathcal{C}))$
  end if
end for
return r_n, $\mathcal{V}$

errors before they enter the vault. **Stage 3 (Verify):** The solver checks $\textsc{Sat}(\mathcal{V} \cup \{\varphi_n\})$. If satisfiable, φ_n is added to the vault and r_n is returned. If unsatisfiable, the solver extracts the minimal unsatisfiable core $\mathcal{C} \subseteq \mathcal{V} \cup \{\varphi_n\}$, identifying the exact beliefs that conflict. **Stage 4 (Repair):** The LLM receives q_n, r_n, and natural language explanations of each belief in $\mathcal{C}$. It generates a revised response addressing the specific conflict. This repeats for up to $k = 3$ iterations.

Algorithm 1 gives the complete procedure.

Formalization. The NL-to-FOL translation uses the LLM in two stages. Stage 1 extracts atomic claims: given "Penguins are birds that cannot fly," the extractor produces `Bird(penguin)` and ¬`CanFly(penguin)`. Stage 2 translates each claim into Z3-compatible FOL expressions with predicates normalized to a shared ontology in the vault. The self-verification loop back-translates the FOL to natural language and compares it to the original. Without this step, CQC drops by 10.2 points (Table 2).

Verification. Z3 checks $\textsc{Sat}(\mathcal{V} \cup \{\varphi_{n+1}\})$ and extracts the minimal unsatisfiable core on failure. This detects transitive contradictions that pairwise checking misses:

$$\varphi_1 : \forall x.\, \text{Bird}(x) \rightarrow \text{CanFly}(x), \quad \varphi_2 : \text{Bird}(\text{penguin}), \quad \varphi_3 : \neg\text{CanFly}(\text{penguin}) \tag{1}$$

No pair is contradictory. The conjunction is unsatisfiable. Only full satisfiability checking finds this.

Targeted repair. The repair module constructs a prompt containing the original query, the conflicting response, natural language explanations of each belief in $\mathcal{C}$, and the specific logical conflict. This is not rejection sampling. The LLM receives diagnostic information about why its response fails and generates a targeted correction.

AGM belief revision. Some contradictions reflect genuine updates. We implement AGM-style belief revision (Alchourrón et al., 1985) with three operations: expansion ($\mathcal{V}+\varphi$, add when consistent), contraction ($\mathcal{V}-\varphi$, remove and dependents via provenance graph), and revision ($\mathcal{V}*\varphi$, contract by $\neg\varphi$ then expand by φ per the Levi identity). The system distinguishes repair from revision using explicit cues, temporal indicators, and confidence scoring.

LogicBench-Cross benchmark. Existing benchmarks (FOLIO, ProofWriter, LogiQA 2.0) evaluate single-query reasoning. None test consistency across a sequence of related queries. We release LogicBench-Cross: 500 scenarios, each with 5–10 related queries, spanning five domains (medical, legal, scientific, commonsense, mathematical) at three difficulty levels (direct negation, transitive multi-hop, subtle domain-specific). Domain experts wrote 50 seed scenarios with ground-truth FOL. LLM-assisted expansion

Table 1: Cross-query consistency on LogicBench-Cross. CQC (%,↑), CR (↓), Acc (%,↑), BCS (↑). †LoCo requires fine-tuning (open-source only).

Model	Method	CQC ↑	CR ↓	Acc ↑	BCS ↑
GPT-4o	Vanilla	34.2	2.41	71.3	0.62
	Self-Consistency	38.1	2.18	73.0	0.65
	Logic-LM	41.5	1.93	76.8	0.68
	BeliefBank	52.3	1.44	72.1	0.74
	LogicVault	**78.6**	**0.53**	**83.1**	**0.92**
Claude 3.5	Vanilla	37.8	2.26	73.1	0.64
	Self-Consistency	41.2	2.04	75.4	0.67
	Logic-LM	44.0	1.81	78.2	0.70
	BeliefBank	54.9	1.38	74.0	0.76
	LogicVault	**80.2**	**0.48**	**84.7**	**0.93**
Llama-3.1-70B	Vanilla	28.4	2.87	64.2	0.55
	Self-Consistency	32.0	2.63	66.8	0.58
	Logic-LM	35.2	2.31	70.1	0.62
	BeliefBank	45.6	1.72	65.3	0.68
	LoCo†	39.1	2.08	68.5	0.64
	LogicVault	**72.3**	**0.71**	**77.8**	**0.89**
Mistral-7B	Vanilla	19.6	3.52	52.1	0.44
	Self-Consistency	22.8	3.31	54.3	0.47
	Logic-LM	25.1	3.04	58.6	0.51
	BeliefBank	33.4	2.41	53.8	0.57
	LoCo†	28.7	2.76	56.9	0.53
	LogicVault	**58.2**	**1.12**	**68.4**	**0.82**

generated 450 more, each verified by two annotators ($\kappa = 0.87$). Adversarial augmentation targeted known LLM failure modes.

We define four metrics. Cross-Query Consistency (CQC): percentage of scenarios with zero contradictions. Contradiction Rate (CR): average contradictions per scenario. Accuracy (Acc): standard single-query accuracy. Belief Coherence Score (BCS): fraction of the vault remaining satisfiable throughout a scenario.

4 Results

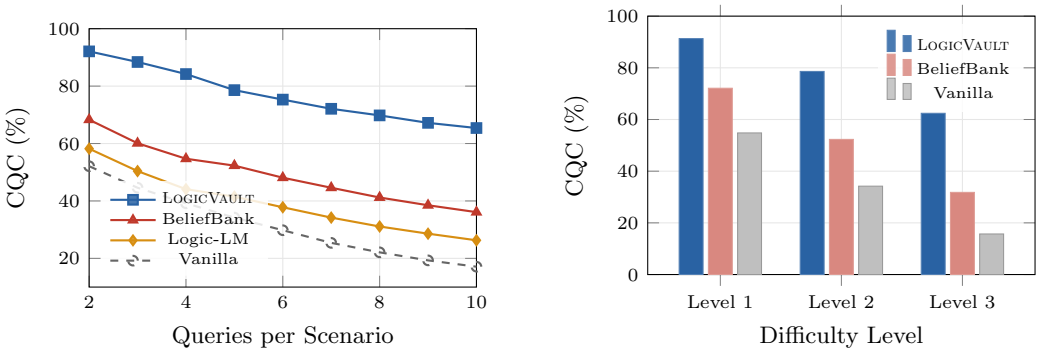
We evaluate six LLMs: GPT-4o (OpenAI, 2024), Claude 3.5 Sonnet (Anthropic, 2024), Llama-3.1-70B (Touvron et al., 2024), Qwen-2.5-72B (Qwen, 2024), Mistral-7B (Jiang et al., 2023), and DeepSeek-R1 (Guo et al., 2025). We compare against five baselines: Vanilla (no checking), Self-Consistency via majority voting over 5 samples (Wang et al., 2022), Logic-LM with per-query solver (Pan et al., 2023), BeliefBank-style constraint propagation (Kassner et al., 2021), and LoCo-style fine-tuning for open-source models (Calanzone et al., 2025). We use Z3 v4.12, $k = 3$ repair iterations, and temperature 0.0.

Table 1 presents the main results on LogicBench-Cross.

LogicVault improves CQC by an average of 2.6× over vanilla across all models. Contradiction rates drop by 68–78%. Single-query accuracy improves by 11–16 points because the consistency constraint prunes incorrect reasoning paths. Mistral-7B with LogicVault (CQC: 58.2) surpasses vanilla GPT-4o (CQC: 34.2). Weak models benefit the most from external consistency enforcement.

Figure 2a shows CQC as a function of query sequence length. Vanilla LLMs degrade rapidly. CQC drops below 20% at 10 queries. LogicVault stays above 65% even at 10 queries. Figure 2b shows CQC by difficulty level. LogicVault's advantage grows at higher difficulty where transitive reasoning is required: a 46.7-point gap at Level 3 versus 36.5 at Level 1.

Table 2 isolates the contribution of each component.



(a) CQC vs. query count (GPT-4o).

(b) CQC by difficulty (GPT-4o).

Figure 2: (a) CQC versus number of queries per scenario. Vanilla GPT-4o drops below 20% at 10 queries. LogicVault stays above 65%. (b) CQC by difficulty level. LogicVault's advantage grows at higher difficulty where transitive reasoning is required.

Table 2: Ablation study on GPT-4o. Each row removes one component from the full system.

Configuration	**CQC ↑**	**CR ↓**	**Acc ↑**	**ΔCQC**
Full LogicVault	**78.6**	**0.53**	**83.1**	–
w/o Repair (reject only)	55.2	0.89	74.3	−23.4
w/o Belief Revision	72.1	0.67	80.4	−6.5
w/o Self-Verification	68.4	0.78	79.1	−10.2
w/o Transitive (pairwise only)	61.8	0.91	78.6	−16.8
w/o Vault (per-query solver)	41.5	1.93	76.8	−37.1

Feeding the LLM precise conflict information produces better corrections than rejection alone. Transitive checking via full SAT solving is the second most important component. Removing the vault entirely reduces the system to per-query Logic-LM, dropping CQC by 37.1 points. LogicVault also improves single-query reasoning as the vault provides accumulated context that helps the LLM reason about new queries more accurately.

Among detected contradictions, 72.4% resolve in the first repair iteration, 21.3% in the second, 4.8% in the third. Only 1.5% require fallback. Average per-query overhead is 1.47s. Z3 handles vaults of 10,000+ propositions in under 1s.

Among remaining failures, we find three causes: formalization failures where nuanced claims resist FOL encoding , repair failures where the LLM cannot produce a consistent alternative, and domain gaps requiring specialized knowledge beyond the vault's expressiveness.

5 Conclusion

We introduced LogicVault, the first framework for cross-query logical consistency in LLMs through persistent symbolic belief states and external solver verification. The system combines inference-time formalization, SMT-based satisfiability checking, targeted repair, and AGM-based belief revision. It reduces cross-query contradictions by 78% and improves single-query accuracy by up to 13% without training. We also release LogicBench-Cross, the first benchmark for this problem.

The primary bottleneck is FOL formalization. Not all natural language claims translate cleanly to first-order logic. The vault grows linearly with queries. Some beliefs are probabilistic and do not fit binary FOL. The framework adds 1.47s average latency per query. Promising directions include integrating probabilistic logic for graded beliefs, learning the formalization mapping via fine-tuning, and applying LogicVault to multi-agent settings with shared vaults.

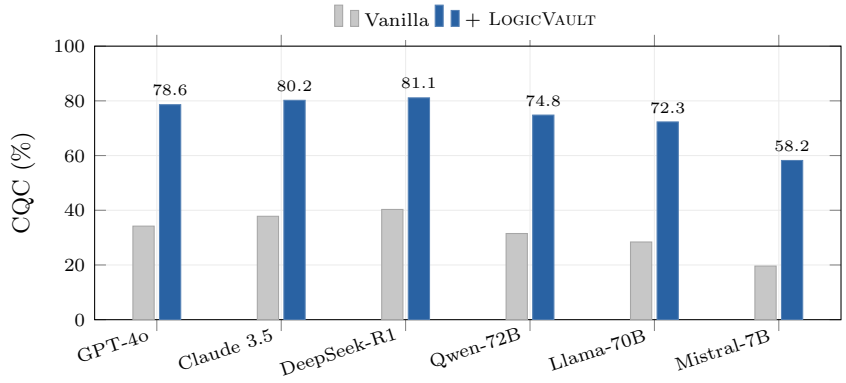


Figure 3: CQC across all six models on LogicBench-Cross. LogicVault provides 2.0–3.0× improvement across model families. Mistral-7B + LogicVault (58.2) exceeds vanilla GPT-4o (34.2).

References

Alchourrón, C. E., Gärdenfors, P., and Makinson, D. On the logic of theory change: Partial meet contraction and revision functions. *Journal of Symbolic Logic*, 50(2):510–530, 1985.

Anthropic. The Claude 3.5 model family. Technical report, 2024.

Calanzone, P., Gerasimova, O., and Valentino, M. LoCo-LMs: Logically consistent language models via neuro-symbolic AI. In *Findings of NAACL*, 2025.

Chen, J., et al. Aristotle: Mastering logical reasoning with a logic-complete decompose-search-resolve framework. In *Proceedings of ACL*, 2025.

de Moura, L. and Bjørner, N. Z3: An efficient SMT solver. In *TACAS*, pp. 337–340, 2008.

Doyle, J. A truth maintenance system. *Artificial Intelligence*, 12(3):231–272, 1979.

Ghosh, S., et al. Logical consistency of large language models in fact-checking. In *Proceedings of ACL*, 2025.

Guo, D., et al. DeepSeek-R1: Incentivizing reasoning capability in LLMs via reinforcement learning. *arXiv preprint arXiv:2501.12948*, 2025.

Han, S., et al. FOLIO: Natural language reasoning with first-order logic. In *Proceedings of AAAI*, 2024.

Jiang, A. Q., et al. Mistral 7B. *arXiv preprint arXiv:2310.06825*, 2023.

Kassner, N., Krojer, B., and Schütze, H. BeliefBank: Adding memory to a pre-trained language model for a systematic notion of belief. In *Proceedings of ACL*, 2021.

Liu, J., et al. LogiQA 2.0: An improved dataset for logical reasoning in natural language understanding. *IEEE/ACM Trans. Audio, Speech, Language Process.*, 2023.

Liu, Z., et al. REPAIR: A universal framework for quantifying compositional logical consistency. In *Findings of NAACL*, 2025.

Mündler, N., He, J., Jenko, S., and Vechev, M. Self-contradictory hallucinations of large language models: Evaluation, detection and mitigation. In *Proceedings of ICLR*, 2024.

Olausson, T. X., et al. LINC: A neurosymbolic approach for logical reasoning by combining language models with first-order logic provers. In *Proceedings of EMNLP*, 2023.

OpenAI. GPT-4o technical report. Technical report, 2024.

Pan, L., Alber, A., Cai, W., and Choi, Y. Logic-LM: Empowering large language models with symbolic solvers for faithful logical reasoning. In *Findings of EMNLP*, 2023.

Qwen Team. Qwen 2.5 technical report. Technical report, Alibaba, 2024.

Tafjord, O., Dalvi, B., and Clark, P. ProofWriter: Generating implications, proofs, and abductive statements over natural language. In *Findings of ACL*, 2021.

Touvron, H., et al. Llama 3: Open foundation models. Technical report, Meta AI, 2024.

Wang, X., et al. Self-consistency improves chain of thought reasoning in language models. In *Proceedings of NeurIPS*, 2022.

Wang, Y., et al. Adaptive symbolic language selection for neuro-symbolic reasoning. *arXiv preprint arXiv:2510.06842*, 2025.

A Prompt Templates

Structured extraction prompt:

```
Given the following response to a factual question, extract all atomic factual
claims as a numbered list.  Each claim should be a single, verifiable statement.
Response:  ‘‘{response}’’
Extract claims:
- Claim 1:  ...
- Claim 2:  ...
```

FOL translation prompt:

```
Translate the following claim into first-order logic using predicates and
constants from this ontology:  {ontology}
Claim:  ‘‘{claim}’’
FOL: ...
Now translate your FOL back to English and verify it matches the original claim.
```

Targeted repair prompt:

```
Your response contradicts your established beliefs.
Question:  ‘‘{query}’’
Your response:  ‘‘{response}’’
Conflict:  You previously stated:
- ‘‘{belief_1_nl}’’
- ‘‘{belief_2_nl}’’
Your new response implies ‘‘{new_belief_nl}’’, which is logically inconsistent
with these.
Provide a revised response consistent with your established beliefs.  Or state
which previous belief should be updated and why.
```

B Extended Results

Table 3: Single-query accuracy (%) across all models with and without LogicVault.

Model	Method	FOLIO	ProofWriter	LogiQA 2.0
GPT-4o	Vanilla	72.3	68.1	58.4
	+ LogicVault	**81.4**	**76.8**	**65.3**
Claude 3.5	Vanilla	74.1	70.3	60.1
	+ LogicVault	**82.8**	**78.2**	**67.0**
DeepSeek-R1	Vanilla	76.2	71.8	62.3
	+ LogicVault	**84.1**	**79.5**	**68.7**
Llama-3.1-70B	Vanilla	65.8	60.4	51.2
	+ LogicVault	**75.6**	**70.1**	**59.8**
Mistral-7B	Vanilla	54.3	48.6	42.1
	+ LogicVault	**65.7**	**59.8**	**52.4**

Table 4: Latency per query (seconds) by vault size, measured on GPT-4o.

Vault Size	**Formalize**	**Verify (Z3)**	**Repair**	**Total**
10	0.81	0.01	0.48	1.30
100	0.82	0.05	0.54	1.41
1,000	0.84	0.18	0.58	1.60
5,000	0.84	0.47	0.61	1.92
10,000	0.85	0.91	0.64	2.40

C LogicBench-Cross Distribution

Table 5: Distribution of LogicBench-Cross scenarios by domain and difficulty.

	Medical	Legal	Scientific	Commonsense	Mathematical	Total
Level 1	34	33	33	34	33	167
Level 2	33	34	34	33	33	167
Level 3	33	33	33	33	34	166
Total	100	100	100	100	100	500