GRACE: Step-Level Benchmark for Faithful Reasoning over Context

Anonymous ACL submission

Abstract

Many reasoning tasks require models to reason over input context, from document-grounded question answering to rule-based deduction. Chain-of-Thought (CoT) prompting produces traces that appear transparent, yet individual steps can silently deviate from the source evidence, even when the final answer is correct. Existing methods detect hallucinations at the response level but fail to identify *where* in the chain a failure occurs or *what type* it is. We introduce **GRACE**[1], the first human-annotated step-level faithfulness benchmark with a data-driven error taxonomy for context-grounded textual reasoning. GRACE covers CoT traces from 10 models across 4 source datasets, with each step annotated for faithfulness, error category, and natural language explanation. A data-driven taxonomy, discovered bottom-up via unsupervised clustering, organizes failures into two tracks: GRACE-Inference (deductive errors) and GRACE-Grounding (factual grounding errors), with four categories each. The evaluation set is human-annotated and challenging by design. Our experiments reveal substantial headroom for current models. In addition, integrating step-level faithfulness signals into reinforcement learning pipelines improves both downstream accuracy and reasoning reliability.

1 Introduction

Chain-of-Thought (CoT) reasoning (Wei et al., 2022; Kojima et al., 2022) is widely employed for knowledge-intensive tasks such as context-augmented multi-hop reasoning and rule-grounded deductive reasoning (Chen et al., 2025). When grounded with reference documents or explicit constraints, models produce step-by-step traces that appear to make reasoning transparent (Trivedi et al., 2023; Lyu et al., 2023). However, a model may extract a fact from context yet draw an invalid conclusion, or produce a plausible inference that no reference supports (Shen et al., 2026; Lanham et al., 2023). From our analysis, a substantial fraction of correct answers are reached through unfaithful intermediate steps, while some fully faithful traces still yield wrong answers. Current evaluation methods can detect that a response contains hallucinations (Niu et al., 2024; ES et al., 2024; Huang et al., 2025; Rawte et al., 2023), but they treat the reasoning chain as a black box, failing to identify *where* the error occurred or *what type* of error it was.

These limitations stem from gaps across three fronts. First, step-level evaluation has matured for mathematical reasoning (Lightman et al., 2024; Wang et al., 2024) and code generation (Zheng et al., 2024), where each step admits formal verification. For textual reasoning, prior work offers automatic step-level metrics (Golovneva et al., 2023) or open-domain verification (Jacovi et al., 2024), but no existing resource provides step-level faithfulness judgments for context-grounded reasoning. A second gap concerns error diagnosis. Output-level faithfulness methods (Niu et al., 2024; ES et al., 2024) produce binary or span-level labels that can flag an unfaithful response but fail to distinguish whether the cause was a logical fallacy or a factual fabrication; no error taxonomy exists for context-grounded reasoning. A third gap is conceptual. Recent work on CoT faithfulness (Shen et al., 2026; Lanham et al., 2023) examines *process faithfulness*: whether a CoT reflects the model's internal decision process. We identify a complementary dimension, *context faithfulness*: whether each step is supported by the source context. These two dimensions are distinct; a step can be process-faithful while being context unfaithful by contradicting the source document, and vice versa. Context faithfulness remains largely unexplored at the step level.

To address these gaps, we introduce **GRACE** (**G**rounded **R**easoning **A**ssessment through **C**hain-of-Thought **E**valuation), a step-level faithfulness benchmark for context-grounded textual reasoning

[1]The code and dataset are available at https://anonymous.4open.science/r/grace-bench-14e8

evaluation. GRACE covers CoT traces from 10 models across 4 source datasets spanning 2 reasoning domains (evidence-grounded and deductive reasoning). Each step carries a rich annotation: a faithfulness label, an error category from our taxonomy, and a natural language explanation grounding the judgment. Rather than imposing a predefined taxonomy, we discover error categories bottom-up: free-form critiques of tens of thousands of unfaithful steps (generated by a strong LLM judge) are embedded and clustered. The result confirms and motivates two empirically distinct failure tracks: **GRACE-Inference** (targeting deductive failures, constraint violations, and logical leaps) and **GRACE-Grounding** (targeting factual contradictions, hallucinated evidence, and extraction errors). To construct the dataset, we employ multiple LLM judges to evaluate each step. Unanimous agreement among judges yields a scalable training set, while human annotation of complex disagreement cases provides a high-quality evaluation set that poses headroom for current models. Beyond evaluation, fine-tuning small models on the training set can match the performance of larger ones. In addition, we show that integrating step-level context faithfulness signal into reinforcement learning (RL) training of downstream tasks improves both task accuracy and reasoning reliability.

Our contributions are threefold: **1) GRACE benchmark**: the first step-level faithfulness benchmark with a data-driven error taxonomy for context-grounded textual reasoning with a high-quality human-annotated evaluation set. **2) Data-driven error taxonomy**: an empirical taxonomy discovered via unsupervised clustering, organized into GRACE-Inference and GRACE-Grounding tracks. **3) Benchmarking and analysis**: The GRACE evaluation set reveals substantial headroom for current models. Furthermore, we demonstrate that integrating context faithfulness signal into RL training improves task accuracy and reasoning reliability.

2 Related Work

2.1 Step-Level Evaluation for Reasoning

Step-level evaluation has been developed primarily for formal domains. In mathematics, PRM800K (Lightman et al., 2024) provides step-level labels for process reward models (PRMs), while MathShepherd (Wang et al., 2024) and OmegaPRM (Luo et al., 2024) scale annotations via outcome verification. For code, OpenCodeInterpreter (Zheng et al., 2024) and MBPP+ (Liu et al., 2023) use execution-based feedback. These domains share deterministic verification, whether a calculation or a code block is correct or not. However, PRMs trained on such domains struggle to generalize to textual reasoning, where errors are semantic and ambiguous (Zeng et al., 2025). Beyond formal domains, recent work extends step-level evaluation: REVEAL (Jacovi et al., 2024) collects step-level human annotations distinguishing attribution and logical correctness against evidence, but labels classify error presence rather than explicit type, and evidence is retrieved post-hoc rather than given to the model as input. VersaPRM (Zeng et al., 2025) scales process supervision to non-math domains via synthetic labeling, but targets step correctness without an error taxonomy. Neither evaluates whether a model faithfully reasons over given input context (see Appendix A for a detailed comparison). GRACE fills this gap with per-step faithfulness labels for context-grounded textual reasoning, and an empirically discovered error taxonomy.

2.2 Hallucination Detection in Context-Grounded Generation

Several approaches target hallucination in context-grounded settings. Claim-extraction methods like FActScore (Min et al., 2023) and RefChecker (Hu et al., 2024) decompose outputs into atomic facts or claim triples checked against the source. Response-level approaches include RAGTruth (Niu et al., 2024) (span-level hallucination types), FAVABench (Mishra et al., 2024) (fine-grained labels), AttrScore (Yue et al., 2023) (three-way attribution error types), and automated metrics such as RAGAS (ES et al., 2024) and ARES (Saad-Falcon et al., 2024). These methods evaluate *what* a model concludes but not *how* it reasons; GRACE targets this gap by evaluating each intermediate reasoning step for faithfulness and error category.

2.3 Chain-of-Thought Faithfulness

A growing body of work examines whether CoT traces reflect the model's internal reasoning. Counterfactual (Lanham et al., 2023; Turpin et al., 2023) and logit-based (Tanneru et al., 2024; Ton et al., 2025) methods probe causal alignment between trace content and model decisions, and FaithCoT-Bench (Shen et al., 2026) formalizes this as instance-level detection. This line of work addresses *process faithfulness*: does the CoT reflect the model's actual reasoning? GRACE targets the

complementary dimension of *context faithfulness*: is each step grounded in the provided context? A model can be fully process-faithful yet context-unfaithful (contradicting the source document).

3 Task Formulation

Context Faithfulness. Consider a set of context references $R = \{r_1, \ldots, r_k\}$, a question q, and a reasoning trace $S = (s_1, s_2, \ldots, s_n)$ produced by a language model conditioned on R and q. We define a step s_i as *context-faithful* if every claim it introduces is either directly supported by the references R, or is a valid logical deduction grounded entirely in R (e.g., by combining facts across passages or applying a rule stated in the context). A step that contradicts, fabricates, or introduces claims unsupported by the provided context is *context-unfaithful*. Context faithfulness is conceptually distinct from *process faithfulness* (Shen et al., 2026): a step can reflect the model's internal reasoning (process-faithful) while being unfaithful to the context.

Evaluation Task. Given references R, question q, and trace S as defined, the evaluator is tasked with outputting a faithfulness label for every step, and an error category and an explanation when the step is unfaithful:

$$f(R,\, q,\, S) \;\longrightarrow\; \big\{\big(y_i,\; g_i,\; e_i\big)\big\}_{i=1}^{n} \qquad (1)$$

where $y_i \in \{\text{FAITHFUL},\ \text{UNFAITHFUL}\}$ is the faithfulness label for step s_i, $g_i \in \mathcal{T}$ is an error category from a finite taxonomy $\mathcal{T}$, and e_i is a free-text explanation. The formulation accommodates different evaluation paradigms: an implementation may classify each step independently (while retaining the preceding trace for linguistic context) or evaluate the entire trace steps jointly.

Challenges. Building a reliable evaluator for this task presents two main difficulties. First, textual reasoning errors are semantic, requiring judgment over natural language meaning, unlike math or code where each step can be checked against arithmetic rules or type systems (Lightman et al., 2024; Wang et al., 2024; Zheng et al., 2024). Second, no step-level ground truth exists for textual reasoning; existing datasets annotate only at the response or span level (Niu et al., 2024; ES et al., 2024; Hu et al., 2024), providing no step-level supervision for context faithfulness. This motivates the construction of **GRACE** benchmark, which provides both step-level annotations and a data-driven error taxonomy for context-grounded textual reasoning evaluation.

4 The GRACE Benchmark

Figure 1 shows the construction pipeline of GRACE, including: data curation and trace generation, bottom-up taxonomy discovery, and data split and human annotation. We detail each stage below.

4.1 Data Curation and Trace Generation

We require source tasks where the answer is derivable from the provided context alone, multi-step reasoning is necessary, and complex arithmetic is absent. Under these criteria we select four datasets spanning two reasoning domains (Table 1). MuSiQue (Trivedi et al., 2022) and 2WikiMHQA (Ho et al., 2020) test multi-hop evidence chaining and entity comparison over passages, representing evidence-grounded reasoning. ReClor (Yu et al., 2020) and LogiQA (Liu et al., 2020) draw from standardized exams (GMAT/LSAT and civil service, respectively) and test deductive reasoning over structured passages. For ReClor and LogiQA, many questions involve hypothetical scenarios not stated in the passage (e.g., "which of the following, if true, would weaken the argument?"), making context faithfulness ill-defined. We initially apply an automatic groundedness filter (using Qwen3.5-27B (Qwen Team, 2026)) that retains only questions answerable from the passage alone, dropping 65% of ReClor and 52% of LogiQA. Dataset details and the filtering prompt are provided in Appendix B.1.

Dataset	Domain	Questions
MuSiQue (Trivedi et al., 2022)	Wiki multi-hop	5,562
2WikiMHQA (Ho et al., 2020)	Wiki comparison	5,000
ReClor (Yu et al., 2020)	GMAT/LSAT exams	1,619
LogiQA (Liu et al., 2020)	Civil service exams	3,525

Table 1: Source datasets for GRACE. Question counts reflect the final pool after groundedness filtering.

We generate single-shot reasoning traces using ten models: eight open-weight from four families (Qwen (Qwen Team, 2025b, 2026), Llama (Meta Llama Team, 2024), Mistral (Mistral AI Team, 2026), Gemma (Farabet and Lacombe, 2026)) and two proprietary (Gemini (Google DeepMind, 2026), GPT (OpenAI, 2024)); the full model list is provided in Appendix B.2. This selection spans 7B to 35B open-weight and proprietary models, ensuring diverse error pattern coverage across capability levels. For each question, we pair it with the source documents and prompt the model to produce step-by-step reasoning with explicit citations (detailed

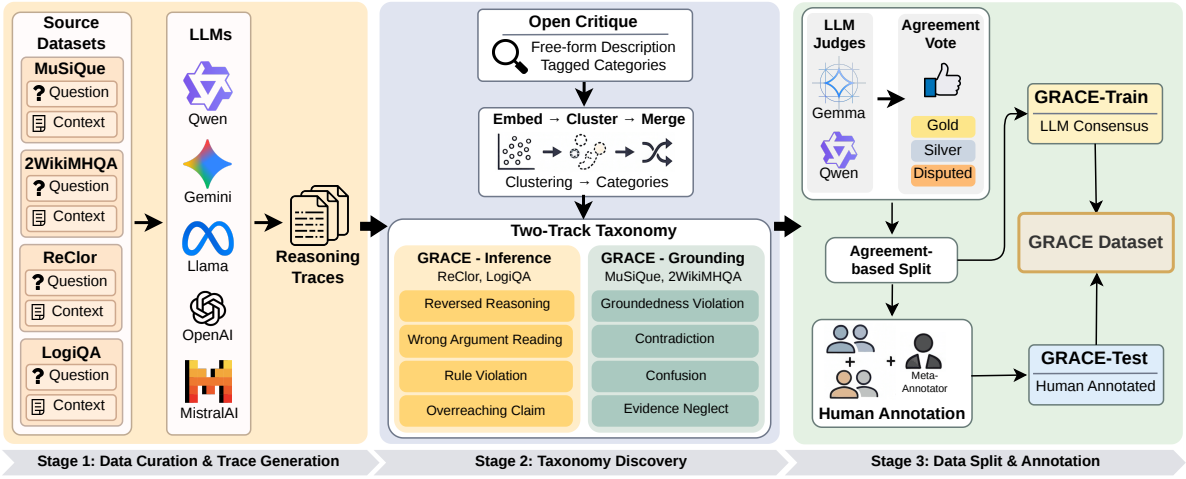

Figure 1: Overview of the GRACE benchmark construction pipeline. Four context-grounded reasoning datasets are curated and CoT traces are generated across model families. Free-form LLM critiques are clustered to discover error categories bottom-up, and then the final taxonomy is manually consolidated, yielding two evaluation tracks. An agreement-based split separates consensus-labeled training data from human-annotated test data.

prompt is in Appendix I). In total, the process produced 160K traces with over 640K steps.

4.2 Taxonomy Discovery

Since no established taxonomy addresses step-level context faithfulness errors, we derive one through a bottom-up process that lets categories emerge from how models actually fail.

Open critique. The goal is not to produce final labels, but to collect raw, unconstrained errors that seed the taxonomy discovery process. We stratify-sample traces across all source models and datasets, then prompt a single LLM evaluator (Qwen3.5-27B (Qwen Team, 2026)) to critique each reasoning step against its source context without predefined categories (detailed prompt is in Appendix I). The evaluator describes what went wrong in a free form description and assigns 1–3 short error tags (e.g., "fabricated entity", "reversed causation"), flagging over 30K steps as unfaithful.

Clustering and consolidation. For each flagged step, we concatenate the error description with its assigned tags into a single representation, embed the 30K combined records with a sentence encoder (Xiao et al., 2023), and cluster them with UMAP (McInnes et al., 2018) + HDBSCAN (McInnes et al., 2017), which assigns 86% of records to one of 138 clusters (the remaining 14% are treated as unclustered noise). An LLM (Qwen3.5-27B) (Qwen Team, 2026) then names each cluster from its representative descriptions and most frequent tags. We then manually validate and merge clusters that share the same underlying error mechanism, consolidating the 138 groups into 8 categories. At this stage, the discovered categories arise from all source models, confirming that they capture general failure modes rather than model-specific artifacts. Details of the process are provided in Appendix C.

Two-track separation. Examining how these categories are distributed across datasets confirms a separation that follows naturally from the underlying task design: in ReClor and LogiQA, at least 98% of errors are reasoning failures, while in MuSiQue and 2WikiMHQA, at least 97% involve mishandling of evidence (Appendix C.4). Multi-hop QA requires extracting and chaining facts from passages, making evidence handling the primary failure mode, whereas logical reasoning tasks require applying deductive rules to given premises, making inference errors dominant. This motivates organizing the taxonomy into two evaluation tracks, each targeting a dimension of context faithfulness.

GRACE-Inference evaluates whether deductive reasoning is faithful to what the context entails, covering ReClor and LogiQA:

- **Reversed Reasoning**: the step inverts a logical or causal relationship, treating a sufficient condition as necessary or swapping antecedent and consequent.
- **Wrong Argument Reading**: the step misidentifies the argument's claim, confusing the conclusion with a premise or targeting a claim the passage does not make.

- **Rule Violation**: the step ignores or breaks a constraint explicitly stated in the passage, including overlooking qualifying exceptions or violating stated conditions.
- **Overreaching Claim**: the step draws a conclusion stronger than the premises support, asserting certainty from possibility or generalizing from a qualified statement.

GRACE-Grounding evaluates whether factual claims are faithful to what the context states, covering MuSiQue and 2WikiMHQA:

- **Groundedness Violation**: the step makes a claim not supported by any reference, through fabrication, injection of external knowledge, or an inference beyond the evidence.
- **Contradiction**: the step directly opposes an explicit statement in the context, including property inversions, numerical mismatches, and negations.
- **Confusion**: the step uses information from the context but attaches it to the wrong entity, swaps attributes, or reverses a stated relationship.
- **Evidence Neglect**: the step claims that information is missing when it is present, or fails to track how entity states change across the narrative.

4.3 Data Split and Annotation

With the traces from Section 4.1 and the taxonomy from Section 4.2, we then assign step-level labels through a multi-stage pipeline.

Multi-judge annotation. Unlike the open critique in Section 4.2, which uses free-form error evaluation, this stage applies the finalized taxonomy to produce structured, reproducible labels over all 160K traces. Each trace is evaluated by three open-weight LLM judges (Gemma-4-27B (Farabet and Lacombe, 2026), Qwen3.5-27B, Qwen3.5-35B-A3B (Qwen Team, 2026)). Each judge receives the source context and all reasoning steps in the trace, along with the taxonomy for the corresponding track (GRACE-Inference or GRACE-Grounding), then assigns a binary faithfulness label and, for unfaithful steps, an error category and free-form explanation (detailed prompt is provided in Appendix I). We aggregate the annotations via majority voting: on binary faithfulness the judges reach unanimous agreement on 79.2% steps; on error category, unanimous agreement reaches 62.1%. From these votes we classify each step into a tier: *gold* (unanimous on both faithfulness and category), *silver* (at least two of three judges agree on both), or *disputed* (insufficient agreement). The trace-level tier is determined by its lowest-quality step.

Split	Traces	Steps	Labels
GRACE-train	6,915	29,612	LLM consensus
GRACE-test	437	2,044	Human-annotated
Total	**7,352**	**31,656**	

Table 2: Data partition splits of GRACE.

Training set. From the full 160K trace pool, the training set draws from the high-agreement subset through two steps: (1) for each of the 8 error categories, we select up to 500 unfaithful traces (50 per source model), drawing from gold traces first and supplementing with silver; (2) we add an equal number of all-faithful traces (gold-only) to achieve a 1:1 faithful-to-unfaithful ratio. The final training set contains 6,915 traces (29,612 steps) with balanced category and model representation (Table 2). We also release the full unfiltered training data to support future research.

Test set. To prevent benchmark saturation, we construct the test set from traces where LLM judges disagree, treating inter-judge disagreement as a proxy for annotation difficulty. We rank traces by a composite difficulty score based on inter-judge disagreement, then select traces while enforcing per-model balance. From the selected traces, we recruit four annotators, all holding bachelor's degrees with knowledge in LLM reasoning, and train them on the GRACE taxonomy. The annotators are divided into two pairs, each assigned a disjoint subset of test traces. Within each pair, both annotators first decide whether the traces are suitable for the task and then independently label every step for faithfulness and error category using the source context and reasoning steps. When the two annotators in a pair disagree, a fifth meta-annotator with research experience in LLM reasoning reviews the conflicting labels and adjudicates after discussion with both parties. Across both tracks, inter-annotator agreement on binary faithfulness reaches Cohen's $\kappa = 0.82$; on error category, $\kappa = 0.75$. The annotation details are described in Appendix H. The final test set contains 437 traces (2,044 steps) (Table 2).

4.4 Dataset Analysis

With the split in Table 2, in this section, we present the empirical properties of the GRACE-test, which includes two tracks: Grounding and Inference.

Unfaithful step rates differ systematically across tracks. Figure 2 shows a consistent gap be-

tween the two tracks: GRACE-Grounding datasets (MuSiQue 44.0%, 2WikiMHQA 49.0%) produce roughly 1.5× more unfaithful steps than GRACE-Inference datasets (LogiQA 29.4%, ReClor 30.8%). One explanation is that multi-hop questions require extracting and linking facts across passages, exposing a potential failure point at every hop, whereas deductive reasoning tasks operate over a single passage where the premises are co-located.

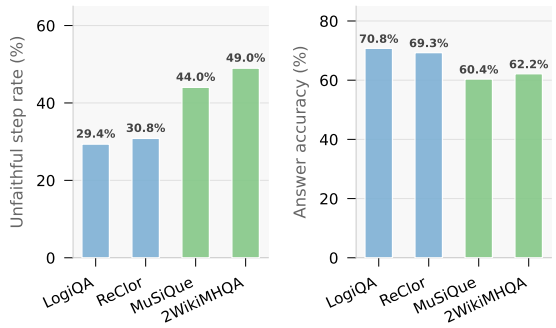

Figure 2: Per-dataset unfaithful step rate (left) and task performance in F_1 (right) in GRACE test.

Correct answers mask unfaithful reasoning. Across the test set, 49.5% of traces that contain at least one unfaithful step still reach the correct final answer (47.3% in GRACE-Inference, 51.6% in GRACE-Grounding). This disconnect between output accuracy and step-level faithfulness is the central motivation for GRACE: output-level evaluation alone cannot distinguish faithful reasoning from reasoning that happens to reach the right answer (we provide case studies in Appendix G). Figure 2 illustrates this at the dataset level. GRACE-Grounding datasets maintain 60–62% F_1 despite unfaithful step rates near 50%, while GRACE-Inference datasets achieve 69–71% F_1 with substantially fewer unfaithful steps. The accuracy gap between tracks (roughly 8 points) is smaller than the unfaithful step rate gap (roughly 16 points), confirming that high unfaithful step rates do not translate proportionally into wrong answers. This implies that accuracy-based evaluation can systematically overestimate reasoning quality, and that step-level faithfulness assessment is necessary to surface errors that outcome metrics conceal.

Error categories reflect distinct failure modes. Figure 3 shows that both tracks are top-heavy but not degenerate. In GRACE-Grounding, *Groundedness Violation* (46.2%) dominates, while in GRACE-Inference, *Overreaching Claim* and *Wrong Argument Reading* share the lead (34.4%

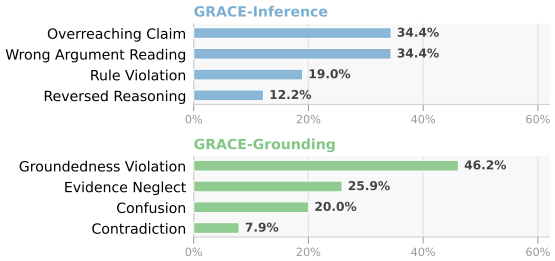

Figure 3: Error category distribution in GRACE test.

each). Most categories exceed 10%, meaning that any detector treating faithfulness as a single binary signal, without category resolution, would miss the diversity of failure.

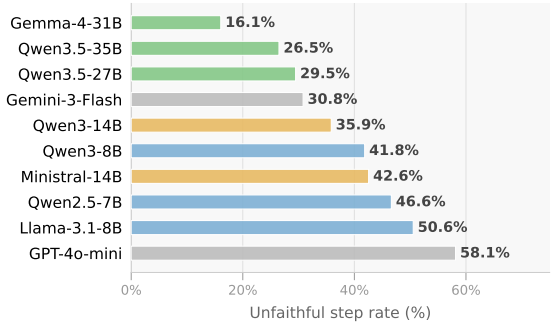

Figure 4: Per-model unfaithful step rate in GRACE test.

Per-model error rates. Figure 4 shows that unfaithful step rate broadly tracks model capability: the three 27–35B open-weight models average 23.8%, while the five 7–14B models average 43.7%. GPT-4o-mini exhibits the highest error rate (58.1%), illustrating that proprietary models do not guarantee context faithfulness. Gemini-3-Flash (30.8%) places between the strong and mid-tier open-weight models, suggesting that the proprietary advantage varies across model families. We provide error category breakdowns in Appendix D.

5 Experiments

We evaluate GRACE along three axes: (1) how well current models detect step-level faithfulness errors (Section 5.2); (2) how inference strategy affects detection quality (Section 5.3); and (3) whether reinforcement learning with process reward can improve faithful reasoning in models (Section 5.4).

5.1 Experimental Setup

Evaluation data. All results are reported on the GRACE test set (437 traces, 2,044 steps).

Metrics. We report two step-level metrics. *Step* F_1 measures binary faithful/unfaithful classification.

Model	GRACE-Grounding				GRACE-Inference				Avg	
	MuSiQue		2WikiMHQA		LogiQA		ReClor			
	Step	Cat	Step	Cat	Step	Cat	Step	Cat	Step	Cat
Gemini-3.1-Pro	78.06	**69.56**	80.53	72.53	**83.28**	**78.77**	**83.97**	**67.44**	**81.46**	**72.08**
GPT-5.4	**80.50**	53.53	82.49	72.81	73.20	52.18	77.92	55.20	78.53	58.43
Gemini-3-Flash	69.47	56.75	81.82	74.86	73.65	59.17	75.82	62.67	75.19	63.36
GPT-5.4-mini	74.12	37.16	69.50	49.49	59.69	33.71	64.08	45.93	66.85	41.57
Gemma-4-31B	72.63	55.47	81.87	62.38	76.69	52.02	78.68	64.13	77.46	58.50
Qwen3.5-35B-A3B	75.50	51.64	**83.08**	60.53	71.12	37.04	73.11	39.40	75.70	47.15
Qwen3.5-27B	67.59	52.10	81.75	76.68	75.76	44.27	73.25	44.39	74.59	54.36
Qwen3-14B	60.92	35.26	62.79	50.32	49.04	25.19	53.66	29.91	56.60	35.17
Qwen3-8B	63.47	31.37	54.88	38.89	42.06	16.43	37.84	22.15	49.56	27.21
Llama-3.1-8B	53.37	20.30	55.11	28.98	41.96	32.05	32.60	26.19	45.76	26.88
Qwen3-8B-SFT	75.68	61.95	78.87	77.22	62.59	54.80	67.70	50.96	71.21	61.23
Qwen3-4B-SFT	71.98	63.31	76.64	72.90	63.30	62.26	62.35	60.55	68.57	64.76
Llama-3.1-8B-SFT	75.74	53.37	79.08	**78.81**	54.61	48.50	55.73	51.42	66.04	58.03

Table 3: Main results on the GRACE test set. Step: step-level binary (faithful/unfaithful) F_1. Cat: error-category macro-F_1 (4-class per track). Best overall per column in bold; best zero-shot underlined.

Category F_1^c measures error categorization among steps labeled unfaithful, computed as macro-F_1 over the categories within each track.

Models. We evaluate on three model groups. *Proprietary*: Gemini-3.1-Pro, Gemini-3-Flash (Google DeepMind, 2026), GPT-5.4 and GPT-5.4-mini (OpenAI, 2026). *Open-weight*: Gemma-4-31B (Farabet and Lacombe, 2026), Qwen3.5 (35B-A3B, 27B) (Qwen Team, 2026), Qwen3 (14B, 8B) (Qwen Team, 2025b), and Llama-3.1-8B (Meta Llama Team, 2024). *GRACE-trained*: Qwen3 (4B, 8B), and Llama-3.1-8B fine-tuned on GRACE-train with LoRA (Hu et al., 2022).

Implementations. Each model receives the full context, question, reasoning trace, and the track taxonomy defined in Section 4.2, then outputs a faithfulness label, error category, and explanation for all steps in a single pass. Detailed inference, training hyperparameters, and prompts are provided in Appendices E and I.

5.2 Main Results

Table 3 presents full results on the GRACE test set. We then highlight the findings as below.

Category classification is harder than binary detection. Across all model groups, the gap between Step F_1 and Category F_1^c is large and consistent. Even the strongest model, Gemini-3.1-Pro, drops from 81.46 to 72.08 across two metrics; for open-weight models the gap widens further (19.0 points for Gemma-4-31B). Binary detection only requires recognizing that *something* is wrong with a step, while category classification demands identifying the specific failure, validating GRACE as a diagnostic benchmark rather than a binary filter.

Frontier models lead but the proprietary/open gap is thin. Gemini-3.1-Pro leads on both metrics (81.46 / 72.08), followed by GPT-5.4 (78.53 / 58.43) and Gemini-3-Flash (75.19 / 63.36). The Step F_1 gap between the best proprietary and best open-weight model (Gemma-4-31B, 77.46) is only 4.0 points, but the Category F_1^c gap reaches 13.6 points (58.50 vs 72.08), indicating that open-weight models are competitive at detecting errors yet lag in fine-grained diagnosis. Model size matters more than access tier: GPT-5.4-mini (66.85 / 41.57) falls below several open-weight models, while performance degrades at smaller scales, with Llama-3.1-8B reaching only 45.76 / 26.88.

SFT on GRACE-train closes the performance gap. Small models fine-tuned on GRACE-train achieve close Step F_1 performance to larger base models and comparable or higher on Category F_1^c. Even Qwen3-4B-SFT achieves (68.57 / 64.76), surpassing Gemini-3-Flash on Category F_1^c and exceeding all open-weight baselines. SFT teaches the taxonomy directly through labeled examples, whereas base models must infer it from a long prompt. The trade-off suggests that smaller models can learn *what* an error is from GRACE-train but still lag behind in the capacity to determine *whether* a given step is erroneous.

5.3 Inference Strategy Comparison

In Table 4 we compare three inference configurations. All strategies receive the context and question, with: *all-steps* (All), where the model additionally receives the full reasoning trace and the error taxonomy, and labels every step in a single pass; *step-by-step* (SbS), where the model receives all preceding steps together with the current step and the taxonomy, and predicts only the current step's label; and *binary* (Bin), which is identical to All but omits the taxonomy and asks the model only to judge each step as faithful or unfaithful.

Model	Grounding		Inference		Avg	
	Step	Cat	Step	Cat	Step	Cat
Gemma-4-31B (All)	**77.3**	58.9	**77.7**	**58.1**	**77.5**	58.5
Gemma-4-31B (SbS)	65.2	**70.2**	69.4	48.7	67.3	**59.5**
Gemma-4-31B (Bin)	69.9	–	67.6	–	68.8	–
Qwen3.5-27B (All)	74.7	64.4	74.5	44.3	74.6	54.4
Qwen3.5-27B (SbS)	75.7	64.7	75.6	48.0	75.6	56.3
Qwen3.5-27B (Bin)	72.1	–	65.3	–	68.7	–
Qwen3-8B-SFT (All)	**77.3**	**69.6**	**65.1**	52.9	**71.2**	**61.2**
Qwen3-8B-SFT (SbS)	77.3	59.0	64.2	**57.4**	70.7	58.2
Llama-3.1-8B-SFT (All)	76.4	66.1	55.2	50.0	65.8	58.0
Llama-3.1-8B-SFT (SbS)	74.2	57.6	51.2	49.9	62.7	53.8

Table 4: Inference strategy comparison: all-steps with taxonomy (All), step-by-step with taxonomy (SbS), and binary without GRACE taxonomy (Bin).

The taxonomy improves detection. We compare All and Bin to isolate the error taxonomy's effect. Gemma-4-31B drops from 77.5 to 68.8 in Avg Step F_1 when the taxonomy is removed, and Qwen3.5-27B drops from 74.6 to 68.7. Providing the taxonomy consistently improves both models, indicating that a structured error framework gives the model useful signals beyond its own knowledge.

Detection and categorization respond differently to inference strategy. No single strategy dominates across models. For Gemma-4-31B, All yields higher Step F_1 (77.5 vs 67.3), but SbS slightly improves Avg Category F_1^c (59.5 vs 58.5), with per-track shifts in opposite directions (Grounding F_1^c: 58.9→70.2; Inference: 58.1→48.7). Conversely, Qwen3.5-27B improves on both metrics under SbS (75.6/56.3 vs 74.6/54.4), benefiting more from reduced input complexity. Thus, inference strategy on base models is model-dependent. In contrast, SFT models, when trained and evaluated under each specific strategy format, favor the All strategy, where models are given the full trace and input context for evaluation.

5.4 Faithfulness-Aware Process Reward

We investigate whether integrating step-level context faithfulness signals into the standard RL training pipeline can improve both task performance and reasoning faithfulness. Table 5 compares Qwen3 (1.7B, 4B) under configurations: base model, GRPO (Shao et al., 2024) with task F_1 reward, and GRPO with F_1 plus a faithfulness-aware PRM (Qwen3-4B trained on GRACE-train). We then report downstream task F_1 on both tracks and a faithfulness score (Faith) averaging faithfulness across trace steps (measured by Qwen3.5-27B). Detailed configurations are in Appendix E.

Model	Strategy	Inference	Grounding	Faith
1.7B	Base	50.7	17.7	67.7
	GRPO (F1 only)	53.5	39.1	63.6
	GRPO (F1 + PRM)	**56.0**	**40.1**	**74.6**
4B	Base	69.5	49.1	77.3
	GRPO (F1 only)	70.2	53.8	77.7
	GRPO (F1 + PRM)	**72.3**	**55.2**	**78.7**

Table 5: GRPO training results. Inference and Grounding report average F1 performance of downstream tasks on respective track datasets. Faith: average step-level faithfulness score measured by Qwen3.5-27B backbone.

Process reward improves both task accuracy and reasoning faithfulness. With task reward alone, the 1.7B model improves F_1 but *drops* Faith from 67.7 to 63.6, revealing a tension between accuracy and faithfulness. Adding PRM resolves this: Faith rises to 74.6 while Inference and Grounding F_1 continue to improve (56.0 / 40.1 vs 53.5 / 39.1 without PRM). At 4B the pattern holds, with PRM yielding gains across all metrics over F_1-only reward. This indicates step-level faithfulness feedback can steer models toward more accurate and faithful reasoning, also validating GRACE-train as an effective signal for process reward models.

6 Conclusion

We presented GRACE, a step-level benchmark for context-grounded textual reasoning, constructed from 10 models across 4 source datasets with labels organized into two tracks and empirically discovered taxonomy. Our experiments show that step-level faithfulness evaluation remains an open problem, with the gap between binary detection and category classification widening as model size decreases. Fine-tuning small models on GRACE-train closes this gap, and integrating step-level faithfulness signal into RL training improves both reasoning performance and reliability.

7 Limitations

GRACE targets unstructured English text across two reasoning domains, providing depth of annotation (step-level labels, error categories, and explanations) over breadth of input modality. Extending coverage to structured inputs (tables, knowledge graphs), multimodal contexts, and multilingual settings is a natural next step. The taxonomy was discovered bottom-up from free-form LLM critiques and consolidated into eight categories through manual inspection by human researchers; the open critique phase uses a single LLM (Qwen3.5-27B) to seed the process, though the final categories reflect human judgment and all evaluation is conducted on the human-annotated test set ($\kappa = 0.82$ on faithfulness, 0.75 on category). The benchmark covers 10 models spanning four families and multiple capability tiers, providing broad error pattern coverage for the current landscape. As model capabilities evolve, the error distribution will naturally shift, requiring periodic updates for representativeness

References

Qiguang Chen, Libo Qin, Jinhao Liu, Dengyun Peng, Jiannan Guan, Peng Wang, Mengkang Hu, Yuhang Zhou, Te Gao, and Wangxiang Che. 2025. Towards reasoning era: A survey of long chain-of-thought for reasoning large language models. *ArXiv*, abs/2503.09567.

Shahul ES, Jithin James, Luis Espinosa Anke, and Steven Schockaert. 2024. Ragas: Automated evaluation of retrieval augmented generation. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2024 - System Demonstrations, St. Julians, Malta, March 17-22, 2024*, pages 150–158. Association for Computational Linguistics.

Clement Farabet and Olivier Lacombe. 2026. Gemma 4: Byte for byte, the most capable open models.

Olga Golovneva, Moya Chen, Spencer Poff, Martin Corredor, Luke Zettlemoyer, Maryam Fazel-Zarandi, and Asli Celikyilmaz. 2023. ROSCOE: A suite of metrics for scoring step-by-step reasoning. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Google DeepMind. 2026. Gemini 3 flash.

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.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Xiangkun Hu, Dongyu Ru, Lin Qiu, Qipeng Guo, Tianhang Zhang, Yang Xu, Yun Luo, Pengfei Liu, Yue Zhang, and Zheng Zhang. 2024. Refchecker: Reference-based fine-grained hallucination checker and benchmark for large language models. *ArXiv*, abs/2405.14486.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2025. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Trans. Inf. Syst.*, 43(2):42:1–42:55.

Alon Jacovi, Yonatan Bitton, Bernd Bohnet, Jonathan Herzig, Or Honovich, Michael Tseng, Michael Collins, Roee Aharoni, and Mor Geva. 2024. A chain-of-thought is as strong as its weakest link: A benchmark for verifiers of reasoning chains. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Bangkok, Thailand. 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 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Haotong Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with pagedattention. *Proceedings of the 29th Symposium on Operating Systems Principles*.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson E. Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, John Kernion, Kamil.e Lukovsiut.e, Karina Nguyen, Newton Cheng, Nicholas Joseph, Nicholas Schiefer, Oliver Rausch, Robin Larson, Sam McCandlish, Sandipan Kundu, and 11 others. 2023. Measuring faithfulness in chain-of-thought reasoning. *ArXiv*, abs/2307.13702.

Hunter Lightman, Vineet Kosaraju, Yuri Burda, Harrison Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2024. Let's verify step by step. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Jian Liu, Leyang Cui, Hanmeng Liu, Dandan Huang, Yile Wang, and Yue Zhang. 2020. Logiqa: A challenge dataset for machine reading comprehension with logical reasoning. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI 2020*, pages 3622–3628. ijcai.org.

Jiawei Liu, Chunqiu Steven Xia, Yuyao Wang, and Lingming Zhang. 2023. Is your code generated by chatgpt really correct? rigorous evaluation of large language models for code generation. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Liangchen Luo, Yinxiao Liu, Rosanne Liu, Samrat Phatale, Harsh Lara, Yunxuan Li, Lei Shu, Yun Zhu, Lei Meng, Jiao Sun, and Abhinav Rastogi. 2024. Improve mathematical reasoning in language models by automated process supervision. *ArXiv*, abs/2406.06592.

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 *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 305–329, Nusa Dua, Bali. Association for Computational Linguistics.

Leland McInnes, John Healy, and Steve Astels. 2017. hdbscan: Hierarchical density based clustering. *Journal of Open Source Software*, 2(11):205.

Leland McInnes, John Healy, Nathaniel Saul, and Lukas Großberger. 2018. Umap: Uniform manifold approximation and projection. *Journal of Open Source Software*, 3(29):861.

Meta Llama Team. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 12076–12100. Association for Computational Linguistics.

Abhika Mishra, Akari Asai, Vidhisha Balachandran, Yizhong Wang, Graham Neubig, Yulia Tsvetkov, and Hannaneh Hajishirzi. 2024. Fine-grained hallucination detection and editing for language models. In *First Conference on Language Modeling*.

Mistral AI Team. 2026. Ministral 3. *Preprint*, arXiv:2601.08584.

Cheng Niu, Yuanhao Wu, Juno Zhu, Siliang Xu, Kashun Shum, Randy Zhong, Juntong Song, and Tong Zhang. 2024. Ragtruth: A hallucination corpus for developing trustworthy retrieval-augmented language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 10862–10878. Association for Computational Linguistics.

OpenAI. 2024. Hello gpt-4o.

OpenAI. 2026. Introducing GPT-5.4. `https://openai.com/index/introducing-gpt-5-4/`.

Qwen Team. 2025a. Qwen2.5 technical report. *Preprint*, arXiv:2412.15115.

Qwen Team. 2025b. Qwen3 technical report. *Preprint*, arXiv:2505.09388.

Qwen Team. 2026. Qwen3.5-omni technical report. *Preprint*, arXiv:2604.15804.

Vipula Rawte, A. Sheth, and Amitava Das. 2023. A survey of hallucination in large foundation models. *ArXiv*, abs/2309.05922.

Jon Saad-Falcon, Omar Khattab, Christopher Potts, and Matei Zaharia. 2024. ARES: an automated evaluation framework for retrieval-augmented generation systems. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), NAACL 2024, Mexico City, Mexico, June 16-21, 2024*, pages 338–354. Association for Computational Linguistics.

John Schulman and Thinking Machines Lab. 2025. Lora without regret. *Thinking Machines Lab: Connectionism*. Https://thinkingmachines.ai/blog/lora/.

Zhihong Shao, Peiyi Wang, Qihao Zhu, Runxin Xu, Junmei Song, Mingchuan Zhang, Y. K. Li, Yu Wu, and Daya Guo. 2024. Deepseekmath: Pushing the limits of mathematical reasoning in open language models. *ArXiv*, abs/2402.03300.

Xu Shen, Song Wang, Zhen Tan, Laura Yao, Xinyu Zhao, Kaidi Xu, Xin Wang, and Tianlong Chen. 2026. Faithcot-bench: Benchmarking instance-level faithfulness of chain-of-thought reasoning. In *The Fourteenth International Conference on Learning Representations*.

Sree Harsha Tanneru, Dan Ley, Chirag Agarwal, and Himabindu Lakkaraju. 2024. On the hardness of faithful chain-of-thought reasoning in large language models. *ArXiv*, abs/2406.10625.

Jean-Francois Ton, Muhammad Faaiz Taufiq, and Yang Liu. 2025. Understanding chain-of-thought in llms through information theory. In *Forty-second International Conference on Machine Learning, ICML*

2025, Vancouver, BC, Canada, July 13-19, 2025, Proceedings of Machine Learning Research. PMLR / OpenReview.net.

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

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2023. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10014–10037, Toronto, Canada. Association for Computational Linguistics.

Miles Turpin, Julian Michael, Ethan Perez, and Sam Bowman. 2023. Language models don't always say what they think: Unfaithful explanations in chain-of-thought prompting. *ArXiv*, abs/2305.04388.

Peiyi Wang, Lei Li, Zhihong Shao, Runxin Xu, Damai Dai, Yifei Li, Deli Chen, Yu Wu, and Zhifang Sui. 2024. Math-shepherd: Verify and reinforce llms step-by-step without human annotations. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 9426–9439. Association for Computational Linguistics.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. 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 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Shitao Xiao, Zheng Liu, Peitian Zhang, and Niklas Muennighoff. 2023. C-pack: Packaged resources to advance general chinese embedding. *ArXiv*, abs/2309.07597.

Weihao Yu, Zihang Jiang, Yanfei Dong, and Jiashi Feng. 2020. Reclor: A reading comprehension dataset requiring logical reasoning. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Xiang Yue, Boshi Wang, Ziru Chen, Kai Zhang, Yu Su, and Huan Sun. 2023. Automatic evaluation of attribution by large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, Singapore. Association for Computational Linguistics.

Thomas Zeng, Shuibai Zhang, Shutong Wu, Christian Classen, Daewon Chae, Ethan Ewer, Minjae Lee, Heeju Kim, Wonjun Kang, Jackson Kunde, Ying Fan, Jungtaek Kim, Hyung Il Koo, Kannan Ramchandran, Dimitris Papailiopoulos, and Kangwook Lee. 2025. VersaPRM: Multi-domain process reward model via synthetic reasoning data. In *Forty-second International Conference on Machine Learning*.

Tianyu Zheng, Ge Zhang, Tianhao Shen, Xueling Liu, Bill Yuchen Lin, Jie Fu, Wenhu Chen, and Xiang Yue. 2024. Opencodeinterpreter: Integrating code generation with execution and refinement. In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, Findings of ACL, pages 12834–12859. Association for Computational Linguistics.

A Detailed Comparison with Related Evaluation Work

We compare GRACE with the closely related benchmarks that provide step-level or response-level evaluation data for reasoning and hallucination detection. Table 6 summarizes the key differences across five benchmarks, grouped by evaluation granularity. Below, we discuss the two step-level benchmarks in detail, as they are most directly comparable to GRACE.

A.1 GRACE vs. REVEAL

REVEAL (Jacovi et al., 2024) is a closely related work, as both provide step-level human annotations for reasoning chains. The two benchmarks differ in two key respects.

Post-hoc retrieved evidence vs. provided context. In REVEAL, the model generates a CoT answer from parametric knowledge alone; evidence is then retrieved from Wikipedia *post-hoc* and given to annotators for verification. A step labeled "unsupported" may therefore reflect a retriever failure rather than a model error, and the dataset is designed primarily to evaluate *verifiers* on given evidence rather than to assess the CoT itself. GRACE evaluates a fundamentally different setting: the model receives all reference documents as input and reasons over them, so every unfaithful step reflects a genuine reasoning failure against the provided context.

Error diagnosis. REVEAL labels steps as correct or incorrect (with a four-way attribution label for factual steps) but does not categorize *what kind of error* occurred. GRACE provides an eight-category error taxonomy enabling diagnostic analysis (e.g., distinguishing reversed logic from entity confusion).

	Step-Level Benchmarks			*Response / Span-Level Benchmarks*	
	REVEAL	**VersaPRM**	**GRACE**	**RAGTruth**	**FAVABench**
Evaluation Design					
Human annotations	Step-level	No (LLM)	Step-level	Span-level	Response-level
Context source	Post Retrieved	None	Provided	Provided	Provided
Error Taxonomy					
# categories	None	None (G/O/B)	8 (two tracks)	4	6
Error Taxonomy Definition	–	–	Bottom-up	Top-down	Top-down
Inference errors	Binary only	Not sep.	Yes (4 cat.)	No	No
Grounding errors	4-way attrib.	Not sep.	Yes (4 cat.)	Yes	Yes

Table 6: Comparison of related evaluation benchmarks. Step-level benchmarks (left) evaluate individual reasoning steps; response/span-level benchmarks (right) evaluate output-level hallucinations. GRACE is the only benchmark combining human-annotated step-level labels, a bottom-up error taxonomy covering both inference and grounding failures. "–" = not applicable; "Not sep." = errors flagged by a single quality label (GOOD/OK/BAD) without distinguishing type.

A.2 GRACE vs. VersaPRM

While REVEAL targets open-domain attribution, VersaPRM (Zeng et al., 2025) shares GRACE's motivation of moving process evaluation beyond mathematics, releasing a multi-domain CoT dataset with step-wise labels across non-math MMLU-Pro domains. Three differences from GRACE:

1. **Step correctness vs. context faithfulness.** VersaPRM checks whether a step is logically correct without evaluating grounding in a provided reference. GRACE evaluates context faithfulness: a step can be logically valid yet unfaithful to the source context.
2. **No error taxonomy.** VersaPRM assigns a single quality score per step (GOOD/OK/BAD). GRACE provides eight diagnostic categories.
3. **Fully synthetic vs. human-verified.** VersaPRM labels are LLM-generated. GRACE's test set is human-annotated.

A.3 Response- and Span-Level Benchmarks

Table 6 also includes RAGTruth (Niu et al., 2024) and FAVABench (Mishra et al., 2024), two benchmarks that evaluate hallucinations in context-grounded generation at the span or response level. Both share GRACE's assumption that context is provided to the model, but evaluate *what* the model concludes rather than *how* it reasons: RAGTruth annotates hallucination spans across four types, and FAVABench provides six fine-grained hallucination labels per response. Neither covers inference errors or provides step-level granularity. GRACE complements them by moving evaluation inside the reasoning chain and adding diagnostic categories for both inference and grounding failures.

B Data Curation Details

B.1 Source Dataset Details

Table 7 lists the four source datasets and their track assignments. Below we describe each dataset's origin, reasoning structure, and curation steps.

Dataset	Raw	Curated	Filter Applied	Track
MuSiQue	19,938	5,562	≥3-hop, answerable	Grounding
2WikiMHQA	167,454	5,000	Stratified sampling	Grounding
ReClor	4,638	1,619	Groundedness	Inference
LogiQA	7,376	3,525	Groundedness	Inference
Total	**199,406**	**15,706**		

Table 7: Curation statistics for each source dataset. The groundedness filter removes questions whose answers require reasoning beyond the provided context.

MuSiQue (Trivedi et al., 2022) is a multi-hop QA dataset constructed from single-hop Wikipedia questions. Each example chains 2–4 facts across paragraphs, with distractors interleaved to prevent shortcuts. We retain only 3-hop and 4-hop answerable questions from the training split (5,562 examples). The multi-paragraph layout fits our chunked reference format, and the multi-hop structure forces faithful traces to cite multiple references.

2WikiMultiHopQA (Ho et al., 2020) pairs Wikipedia articles with Wikidata triples to form multi-hop comparison and bridge reasoning questions across four types: bridge comparison, direct comparison, compositional fact chaining, and transitive inference. We stratify-sample across all four types (2,000 bridge comparison with ≥4 hops; 1,000 each for the rest), producing 5,000 examples. Each includes supporting and distractor paragraphs, requiring selective evidence use.

ReClor (Yu et al., 2020) draws from GMAT and LSAT logical reasoning sections. Each question presents a short argumentative passage and asks the model to reason over its content: identifying conclusions, recognizing flawed inferences, or assessing argument strength. Although all questions are passage-based, many ask the model to evaluate hypothetical scenarios not stated in the passage (e.g., "which of the following, if true, would weaken the argument?"). Since GRACE evaluates context faithfulness, we apply an LLM-based groundedness filter in combination with human verification on the test set to filter questions whose answers depend on reasoning beyond the provided passage, dropping 65% of the training and leaving 1,619 context-grounded samples. Detailed prompt template can be found at Appendix I.

LogiQA (Liu et al., 2020) consists of logical reasoning problems from Chinese civil service examinations, translated to English. Each question provides a passage containing premises or situational descriptions, and asks the model to draw deductive inferences, apply conditional reasoning, or evaluate categorical syllogisms based on the passage content. We apply a minimum context length of 200 characters and the same groundedness filter used for ReClor, together dropping 52% of the training split and yielding 3,525 examples.

B.2 Source Models and Generation Configuration

To ensure that the resulting error taxonomy reflects general failure modes rather than architecture-specific artifacts, we select models spanning four open-weight families (Qwen (Qwen Team, 2025a,b, 2026), Llama (Meta Llama Team, 2024), Mistral (Mistral AI Team, 2026), Gemma (Farabet and Lacombe, 2026)) and two proprietary families (OpenAI (OpenAI, 2024), Gemini (Google DeepMind, 2026)). The selection for open-weight models covers a range from 7B to 35B parameters. Table 8 lists all ten models with their sampling parameters. Open-weight models are served via vLLM (Kwon et al., 2023) using the values recommended in each model's official documentation; proprietary models are accessed through batch APIs. All models generate up to 4,096 tokens per trace.

Model	Family	T	Top-p	Top-k
Qwen2.5-7B-Instruct	Qwen	0.7	0.8	20
Qwen3-8B	Qwen	0.7	0.8	20
Qwen3-14B	Qwen	0.7	0.8	20
Qwen3.5-27B	Qwen	0.7	0.8	20
Qwen3.5-35B-A3B	Qwen	0.7	0.8	20
Llama-3.1-8B-Instruct	Llama	0.6	0.9	20
Ministral-3-14B	Mistral	0.7	0.95	20
Gemma-4-31B-IT	Gemma	1.0	0.95	20
GPT-4o-mini	OpenAI	0.7	0.95	–
Gemini-3-Flash	Gemini	0.7	0.95	–

Table 8: Source models and sampling parameters for trace generation. "–" indicates the parameter is disabled or not exposed.

C Taxonomy Discovery Details

This appendix details the taxonomy discovery pipeline from Section 4.2.

C.1 Pipeline Configuration

The open critique phase evaluates a stratified sample: for each of the 10 source models, we include all traces with incorrect final answers plus 2,000 randomly sampled correct-answer traces. The evaluator (Qwen3.5-27B) identifies over 30K steps as unfaithful; the remaining steps are excluded from clustering. We embed the 30K error descriptions using BGE (Xiao et al., 2023), reduce to 10 dimensions with UMAP (McInnes et al., 2018) using 15 neighbors, and cluster with HDBSCAN (McInnes et al., 2017) using a minimum cluster size of 50 and a minimum samples parameter of 10. Of these, ∼26K (86%) are assigned to one of 138 clusters; the remaining ∼4.3K (14%) are HDBSCAN noise.

C.2 Cluster-to-Category Mapping

Table 9 shows the 20 largest clusters from the global pass. Grounding-type errors are well-differentiated: rows 6, 7, 9, 16, 19, and 20 each describe entity or attribute mix-ups in different knowledge domains, yet all map to *Confusion*. Similarly, rows 3, 8, 10–12, and 15 all describe unsupported claims across different domains, mapping to *Groundedness Violation*. Cluster 1 ("Formal Logical Fallacy") contains all inference-type errors as a single undifferentiated group and is further decomposed via a focused second pass (Section C.3).

C.3 Inference-Track Sub-Categorization

Free-form descriptions of inference errors tend to be generic ("the conclusion does not follow"), so the global clustering collapsed them into a single cluster. To achieve the same granularity as the GRACE-Grounding categories, we re-ran the pipeline on the 11K deductive-error steps from LogiQA and ReClor with a prompt that extracts log-

#	Discovered Cluster Name	Size	→ Category
1	Formal Logical Fallacy	10.7K	*see Section C.3*
2	Direct Context Contradiction	3.4K	Contradiction
3	Fabricated Facts	2.7K	Groundedness Violation
4	Invalid Logical Leaps	1.5K	Groundedness Violation
5	Incorrect Relationship Inference	1.2K	Confusion
6	Genealogical Relationship Confusion	338	Confusion
7	Historical Anachronism	333	Confusion
8	External Knowledge Injection	315	Groundedness Violation
9	Geographic and Entity Conflation	295	Confusion
10	Fabricated Ranking from Lists	282	Groundedness Violation
11	Unsubstantiated Deduction	261	Groundedness Violation
12	Hallucinated Administrative Capital	221	Groundedness Violation
13	Character Knowledge Mismatch	203	Evidence Neglect
14	Geographic Contradiction	201	Contradiction
15	Fabricated Historical Relationships	194	Groundedness Violation
16	Misattribution of Entity Attributes	192	Confusion
17	False Negatives on Explicit Data	192	Evidence Neglect
18	False Negatives on Explicit Context	191	Evidence Neglect
19	Entity and Location Conflation	185	Confusion
20	Geographical Cultural Conflation	185	Confusion

Table 9: Top 20 clusters (of 138) from the global clustering pass, mapped to final categories by error mechanism.

ical structure (premise form, operation attempted, error pattern). HDBSCAN discovered 53 subclusters, which we consolidated into 4 categories:

- **Rule Violation** (28.0%) - ignores a stated constraint or condition.
- **Reversed Reasoning** (26.4%) - inverts a logical or causal relationship.
- **Overreaching Claim** (24.4%) - concludes beyond what evidence supports.
- **Wrong Argument Reading** (21.2%) - misidentifies what the argument claims.

The roughly uniform distribution confirms that each category captures a distinct failure mechanism rather than a single dominant mode.

C.4 Cross-Dataset Error Distribution

Table 10 shows the split between GRACE-Inference and GRACE-Grounding errors across datasets during the taxonomy discovery phase.

Dataset	Inference	%	Grounding	%
LogiQA	6.6K	99.1	59	0.9
ReClor	4.1K	98.6	59	1.4
MuSiQue	380	3.0	12.1K	97.0
2WikiMHQA	3	0.1	2.7K	99.9

Table 10: GRACE-Inference vs. GRACE-Grounding error split per dataset (discovery phase). The sharp separation (≥97%) motivated the two-track design.

D Additional GRACE Dataset Analysis

This appendix expands on the dataset analysis in Section 4.4, examining how error patterns distribute across source models.

Error category prevalence varies by model scale and track. Figure 5 breaks down error categories by source model. Two patterns stand out. First, *Groundedness Violation* is the most frequent GRACE-Grounding error for most source models. Even Gemma-4-31B, the model with the lowest overall error rate (16.1%), still produces grounding errors. This persistence across model sizes indicates that scaling alone does not eliminate unsupported claims in multi-hop evidence tasks. In contrast, the GRACE-Inference categories show clearer differentiation by model size: smaller models (Qwen3-8B, Qwen2.5-7B, Llama-3.1-8B) produce elevated counts across all four categories, while the 27–35B models show lower counts.

Second, the heatmap reveals model-specific variation beyond what scale alone would predict. Ministral-14B stands out with an unusually high *Evidence Neglect* count (20) relative to its tier, while Qwen3.5-27B produces a disproportionately high *Groundedness Violation* count (25) despite its scale, surpassed only by the much smaller Llama-3.1-8B (30). This variation across models of similar size supports sampling traces from a diverse model pool when constructing a benchmark.

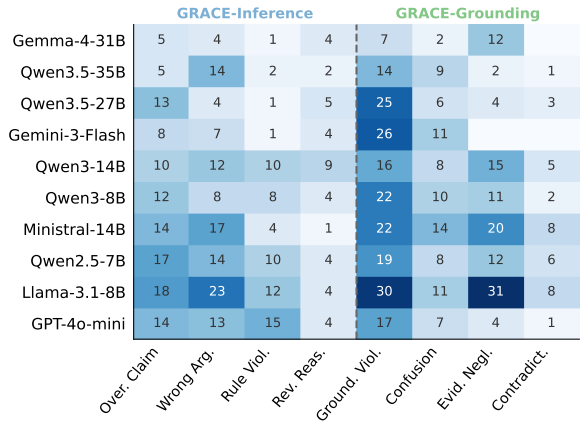

Figure 5: Error category distribution across source models. Each cell shows the number of test traces in which the model produced that error type. Models are sorted by overall unfaithful step rate (lowest at top).

The two tracks occupy structurally distinct reasoning regimes. Figure 6 characterizes the structural diversity of GRACE traces along four dimensions. GRACE-Inference traces reason over short, focused contexts (median 128 words) and produce longer reasoning chains (median 5 steps, 236 words total), reflecting the multi-step deductive structure of logical reasoning tasks where the model must build an argument across several inferential moves.

GRACE-Grounding traces operate over substantially longer multi-passage contexts (median 1,562 words) with shorter chains (median 4 steps, 127 words), as multi-hop QA requires extracting and linking facts rather than extended deduction.

Panel (d) makes this contrast explicit: the two tracks form largely non-overlapping clusters in the context length vs. trace length space. This structural separation has a practical consequence for evaluation. A faithfulness detector that succeeds on short-context logical arguments (where the entire reference fits in a few sentences) may fail on long multi-passage contexts (where relevant evidence is distributed across paragraphs and must be retrieved before it can be verified). The presence of both regimes within a single benchmark ensures that evaluation models cannot rely on a single verification strategy.

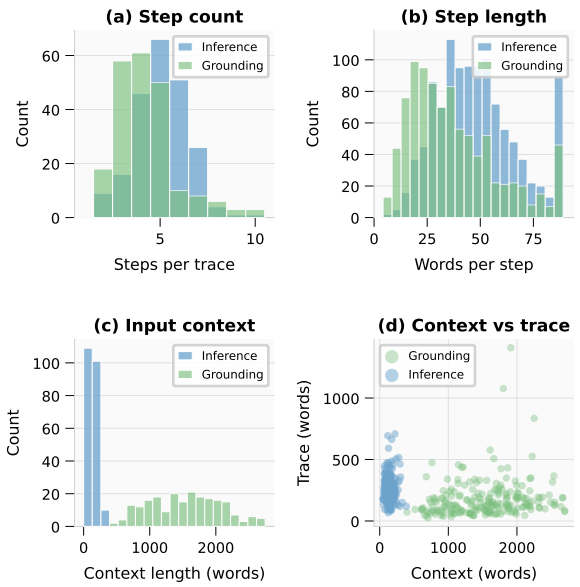

Figure 6: Trace structure analysis of the GRACE test set by track. (a) Step count per trace. (b) Word count per individual step. (c) Input context length. (d) Context length vs. total trace length, showing two non-overlapping reasoning regimes.

E Additional Experiment Details

E.1 SFT Training

We fine-tune on GRACE-train using standard next-token cross-entropy. Given a training sample with context c, question q, and target sequence $y = (y_1, \ldots, y_T)$ comprising step-level reasoning and a final answer, the loss is:

$$\mathcal{L}_{\text{SFT}}(\theta) = -\frac{1}{T}\sum_{t=1}^{T} \log\, p_\theta(y_t \mid y_{<t},\, c,\, q)$$

The model learns to produce faithful step-by-step reasoning by imitating the annotated traces in GRACE-train.

E.2 GRPO Training

In Section 5.4, we optimize the policy with Group Relative Policy Optimization (GRPO) (Shao et al., 2024). For each prompt, G completions are sampled, and the advantage of the i-th response within group g is normalized relative to the group:

$$A_i^{\text{GRPO}} = \frac{R_i - \bar{R}_g}{\sigma_g + \epsilon}$$

where $\bar{R}_g$ and σ_g are the mean and standard deviation of rewards in group g. The policy is updated with a clipped objective:

$$\mathcal{L}_{\text{GRPO}}(\theta) = -\mathbb{E}\big[\min\big(\rho_t\, A_t,\ \text{clip}(\rho_t,\, 1\pm\epsilon_c)\, A_t\big)\big]$$

where $\rho_t = \pi_\theta / \pi_{\theta_{\text{old}}}$ is the importance ratio and ϵ_c is the clipping bound.

Reward design. Each completion receives a reward composed of a format penalty, an answer F_1 score, and optionally a process reward:

- **Format reward.** $R_{\text{fmt}} = 0$ if the completion contains at least one reasoning step ("Step k:") and a final answer; -1 otherwise.
- **Answer F_1 reward.** The predicted answer is compared with the gold answer using word-level F_1: $R_{F_1} = F_1(\hat{a},\ a^*) \in [0, 1]$.
- **Process reward.** A Qwen3-4B model fine-tuned on GRACE-train predicts a binary faithfulness label $s_k \in \{0, 1\}$ for each reasoning step, where s_k=1 indicates faithful and s_k=0 unfaithful. The process reward is the fraction of faithful steps: $R_{\text{proc}} = \frac{1}{K}\sum_{k=1}^{K} s_k$.

We compare two configurations:

$$F_1\ \ only : R = R_{\text{fmt}} + R_{F_1}$$
$$F_1 + PRM : R = R_{\text{fmt}} + R_{F_1} + R_{\text{proc}}$$

Evaluation. The purpose of these experiments (from Section 5.4) is to determine whether incorporating step-level context faithfulness signals into the reinforcement learning pipeline improves both downstream task performance and reasoning faithfulness. We measure task accuracy by reporting the answer F_1 score on the GRACE-Inference and GRACE-Grounding tracks. To evaluate reasoning quality, we compute an overall faithfulness score (Faith), which averages the context faithfulness of each step in the generated reasoning traces, as measured by a Qwen3.5-27B judge model. Comparing

policies trained solely with the answer F_1 reward against those augmented with our process reward model isolates the effect of step-level faithfulness feedback.

E.3 Training Configuration

All models use LoRA (Hu et al., 2022) adapters on all linear layers with AdamW (Loshchilov and Hutter, 2019), following the suggestions of Schulman and Lab (2025). Training is conducted on a single server with 4× NVIDIA A6000 GPUs.

For SFT, we trained Qwen3-8B, Qwen3-4B, Llama-3.1-8B, and Llama-3.2-3B on GRACE-train for 3 epochs.

For GRPO, we trained Qwen3-1.7B and Qwen3-4B for 2,000 steps on the four source datasets (MuSiQue, 2WikiMHQA, ReClor, LogiQA). We then measure their downstream task F_1 performance and also step faithfulness using a larger backbone (Qwen3.5-27B). Table 11 lists the full hyperparameters.

Hyperparameter	SFT	GRPO
LoRA rank (r)	128	16
LoRA alpha (α)	32	32
Target modules	all-linear	all-linear
Learning rate	1e-4	5e-5
Effective batch size	16	16
Training duration	3 epochs	2,000 steps
Training data	GRACE-train	Source datasets

Table 11: Training hyperparameters for SFT and GRPO, following the suggestions of Schulman and Lab (2025).

E.4 GRPO Training Dynamics

Figure 7: Answer F_1 during GRPO training for the 4B model.

Figure 7 shows the Answer F_1 reward during training for the two GRPO configurations on the 4B model (F_1 reward only and F_1 + faithfulness-aware PRM). Both variants follow the same ascent during the first ∼400 steps, then diverge: the F_1-only curve plateaus around 0.62–0.64, while the PRM variant breaks through to 0.65–0.66 and sustains a 1–2 point margin. The plateau in the F_1-only run indicates that outcome-level reward has been exhausted; PRM continues to provide gradient signal on individual steps after trace-level F_1 saturates, which is consistent with the final evaluation gaps in Table 5. This suggests that process reward is more valuable in later training stages, and combining it with outcome reward does not interfere with the initial learning phase.

F Error Category Examples

In tables 12 and 13, we present one representative example per error category from GRACE-Logic and GRACE-Evidence, respectively. For each example we show a relevant excerpt from the source context, the model's reasoning step verbatim, and a brief explanation of the error.

G Case Study: Correct Answers Despite Unfaithful Reasoning

As noted in Section 4.4, 49.5% of traces containing at least one unfaithful step still reach the correct final answer. To illustrate how this can occur, we present two scenarios observed in the test set.

Scenario 1: Knowledge Shortcut and Injection. The model falls back on memorized world knowledge during an intermediate reasoning step, bypassing the provided context. Because the injected knowledge happens to align with the gold label, the final answer is correct despite the unfaithful step. Figure 8 presents an example from GRACE-Grounding.

Scenario 2: Self-Correction. The model produces unfaithful reasoning steps, then partially recognizes the issue in a subsequent step and adjusts its conclusion. The corrective step is itself faithful and leads to the correct final answer, but the preceding errors remain in the trace. Figure 9 presents an example from GRACE-Inference.

Both scenarios show that a correct final answer alone cannot reveal whether the underlying reasoning is faithful; step-level assessment, as provided by GRACE, is necessary to surface such cases.

H Annotation Interface

Figure 10 shows the web-based annotation interface used during human annotation (Section 4.3). The interface is organized into three panels. The left panel presents the source context with numbered references and the question with its gold answer highlighted. The center panel displays the model's reasoning chain as a sequence of numbered steps, each tagged with its current annotation status (e.g., *pending*). The right panel contains the annotation form for the currently selected step.

For each step, the annotator first assigns a binary faithfulness label (*Faithful* or *Unfaithful*). When *Unfaithful* is selected, the form reveals the error categories from the relevant track for annotators to select the proper category. The annotator then writes an explanation of the error and rates their own confidence on a 5-point scale. An optional difficulty tag and a "flag for review" checkbox are available for cases that warrant a second opinion.

Each annotator is assigned a set of traces. Step-level tabs at the top of the annotation panel allow navigating across steps within a trace. A progress counter tracks how many steps have been submitted. If a trace is unintelligible or malformed, the annotator may skip it via the "Skip Trace" button.

I Prompt Templates

For better reproducibility, we present all prompt templates in the appendix as bellow:

- Figure 11: Prompt template for trace generation. Models reason step-by-step grounded in provided references and cite relevant sources at each step.
- Figure 12: Prompt template for groundedness classification. Used to filter logic questions that require external hypothetical facts.
- Figure 13: Prompt template for open critique phase. Evaluators freely describe reasoning errors without a predefined taxonomy, producing unbiased descriptions for taxonomy discovery.
- Figure 14: Prompt template for GRACE steps annotation. Judges classify the reasoning error of all steps to predefined categories.

Category	Example
Reversed Reasoning Llama-3.1-8B ReClor	**CONTEXT:** *"[. . .] to have the political impact necessary to influence legislation, a group must be united."* **STEP:** "This inability to unite is necessary for a group to have the political impact necessary to influence legislation, implying that a large group may become ineffective due to internal conflicts." **ERROR:** The passage requires that a group *must be united*; the step inverts this into "inability to unite is necessary," reversing the direction of the logical requirement.
Wrong Argument Reading Qwen2.5-7B LogiQA	**CONTEXT:** *"[. . .] the second anti-French [alliance] composed of Britain, Russia, Austria and other countries [. . .] fought against Napoleon."* **STEP:** "Despite being part of this alliance, Napoleon managed to surprise everyone by crossing into Italy and defeating the Austrian army." **ERROR:** The passage describes an *anti*-French alliance that fought *against* Napoleon. The step misreads him as a member of the alliance rather than its target.
Rule Violation Qwen2.5-7B LogiQA	**CONTEXT:** *"[. . .] Berto does not go to Canada or the UK."* **STEP:** "With Dampy going to the Netherlands and Berto going to Canada, the only remaining destination for Bill is the UK." **ERROR:** The passage explicitly excludes Canada for Berto. The step assigns him there anyway, violating a stated constraint.
Overreaching Claim Qwen2.5-7B ReClor	**CONTEXT:** *"Almost all advances in genetic research give rise to ethical dilemmas."* **STEP:** "The references state that almost all advances in genetic research give rise to ethical dilemmas. This means that every advance in genetic research is associated with an ethical dilemma." **ERROR:** The passage qualifies with "almost all." The step silently upgrades this to "every," overgeneralizing a qualified statement into an absolute claim.

Table 12: Representative examples from **GRACE-Inference**. Each row shows a context excerpt, the model's reasoning step, and an error explanation.

Category	Example
Groundedness Violation Qwen3.5-35B-A3B MuSiQue	**CONTEXT:** *"Rabid Dogs is an Italian film directed by Mario Bava [. . .]" "The Embassy of Poland in Rome [. . .]"* [No reference states where Bava died.] **STEP:** "The references do not explicitly state where Mario Bava died. However, the question implies the director died in Rome, since the only Embassy of Poland mentioned is in Rome and the film is Italian." **ERROR:** The model explicitly acknowledges the gap in the context but fills it with an unsupported inference. No reference links Bava's death to Rome.
Contradiction Qwen3-14B MuSiQue	**CONTEXT:** *"[. . .] the Battle of Borodino resulted in approximately 44,000 Russian and 35,000 French dead [. . .] Although the French had won, the Russian army had accepted, and withstood, the major battle Napoleon had hoped would be decisive."* **STEP:** "The people who won the Battle of Borodino were the Russians, as the reference explicitly states that the Russians 'had accepted, and withstood, the major battle Napoleon had hoped would be decisive."' **ERROR:** The context explicitly states "the French had won." The step misinterprets a quote about Russian resilience as evidence of Russian victory, directly contradicting the passage.
Confusion Llama-3.1-8B MuSiQue	**CONTEXT:** *"Hogeschool is used in Belgium and in the Netherlands [. . .] similar to the ammattikorkeakoulu in Finland."* **STEP:** "The term used in Country A (Finland) and the country the Women's Euro 2017 is being held (Netherlands) to refer to an institution like a German Fachhochschule is hogeschool." **ERROR:** The context states that Finland uses "ammattikorkeakoulu" while the Netherlands uses "hogeschool." The step assigns "hogeschool" to both countries, mixing up which country uses which term.
Evidence Neglect Qwen3-14B MuSiQue	**CONTEXT:** *"Circus Maximus [. . .] Built in Old Kingdom era [. . .]"* **STEP:** "The references do not explicitly mention when the Circus Maximus was built, so we cannot determine this from the provided information." **ERROR:** The context explicitly states "Built in Old Kingdom era." The model claims this information is absent when it is directly provided.

Table 13: Representative examples from **GRACE-Grounding**. Format follows Table 12.

Scenario 1: Knowledge Shortcut and Injection

Qwen3.5-27B · MuSiQue · GRACE-Grounding | **Final answer: Stockholm Arlanda Airport ✓**

QUESTION: *What is the main international airport in the city where Daniel Majstorovic's team is headquartered?*

PASSAGES (EXCERPTS):

[ref_1] Carlos Banda began coaching youth teams of IF Brommapojkarna and moved on to Stockholm rival Hammarby IF in 2003.

[ref_2] Stockholm Arlanda Airport is an international airport located in the Sigtuna Municipality of Sweden, 37 km north of Stockholm. . .

[ref_5] Daniel Majstorović started his career with IF Brommapojkarna in his native Sweden. After a year, he moved to SC Fortuna Köln. . .

Step 1 UNFAITHFUL · *Groundedness Violation*

". . . [ref_1] mentions Carlos Banda coaching IF Brommapojkarna and then moving to Hammarby IF in Stockholm. This establishes a connection between IF Brommapojkarna and Stockholm. **However, IF Brommapojkarna is actually based in Bromma, a district of Stockholm. Malmö FF is based in Malmö.** . . . Therefore the city is Stockholm."

Annotation: The claims "Bromma is a district of Stockholm" and "Malmö FF is based in Malmö" are correct world knowledge but appear nowhere in the provided references. The model bypasses the context and relies on memorized geographic facts to resolve an ambiguity the passages do not address.

Step 2 FAITHFUL

"Stockholm Arlanda Airport is the largest airport in Sweden and the major gateway to international air travel, located in Stockholm County [ref_2]."

Step 3 FAITHFUL

"Daniel Majstorović's career began with IF Brommapojkarna, which is based in Stockholm. The main international airport serving Stockholm is Stockholm Arlanda Airport [ref_1, ref_2]."

The model reaches the correct answer, but Step 1 introduces external knowledge (geographic facts about Bromma and Malmö) that is absent from the references. The correct answer is an artifact of the shortcut aligning with ground truth, not of faithful context-grounded reasoning.

Figure 8: Case study: **Knowledge Shortcut and Injection**. The model injects memorized world knowledge (highlighted in bold) to resolve an ambiguity that the provided references do not address, arriving at the correct answer through a reasoning shortcut rather than faithful context-grounded inference.

Scenario 2: Self-Correction

Qwen3-8B · ReClor · GRACE-Inference **Final answer: (B) ✓**

QUESTION: *The application of the principle is most vulnerable to criticism on the grounds that*

OPTIONS:

(A) the application fails to establish whether Wilton was given the opportunity to repair the brakes

(B) asserting something without justification is not the same as asserting something one knows to be false

(C) Harris might not have believed Wilton's statement about the bicycle's condition

(D) the application uses the word "defective" in a sense that is crucially different from how it is used in the statement of the principle

PASSAGE:

Principle: If you sell an item that **you know to be defective**, telling the buyer that the item is sound, you thereby commit fraud. **Application:** Wilton sold a used bicycle to Harris, **knowing very little about its condition**. Wilton told Harris that the bicycle was in good working condition, but Harris soon learned that the brakes were defective. Wilton was therefore guilty of fraud.

Step 1 UNFAITHFUL · *Wrong Argument Reading*

"The principle states that if someone sells an item they know to be defective while falsely claiming it is sound, they commit fraud [ref_1]. In the application, Wilton sold a used bicycle, knew little about its condition, and told Harris it was in good working condition, but the brakes were later found to be defective. **This scenario fits the principle's description of fraud** because Wilton misrepresented the item's condition, and the item was indeed defective."

Annotation: The principle requires that the seller *know* the item is defective. The passage states Wilton "knew very little about its condition," so the application does not satisfy the principle's knowledge requirement. The step misreads this gap and accepts the application as valid.

Step 2 FAITHFUL · *Self-Correction*

"The application's vulnerability lies in the assumption that Wilton knew the brakes were defective. **However, the reference only states that Wilton 'knew very little about its condition,'** which does not necessarily mean Wilton knew the brakes were defective. This distinction is crucial because the principle requires knowledge of the item's defectiveness to establish fraud. The application assumes knowledge of the defect, which is not explicitly supported by the reference."

Step 1 treats the application as satisfying the fraud principle, overlooking that Wilton lacked knowledge of the defect. Step 2 explicitly reverses this reading ("However, the reference only states… "), recognizing the gap between "knowing very little" and "knowing to be defective." The final answer is correct because the model corrected its own misreading, but Step 1 remains unfaithful.

Figure 9: Case study: **Self-Correction**. The model initially misreads the argument as satisfying the fraud principle (Step 1), then in Step 2 explicitly contradicts itself and correct the error and then derives to the correct conclusion. Bold text highlights the erroneous claim and its correction. The correct answer results from the model reversing its own unfaithful reasoning.

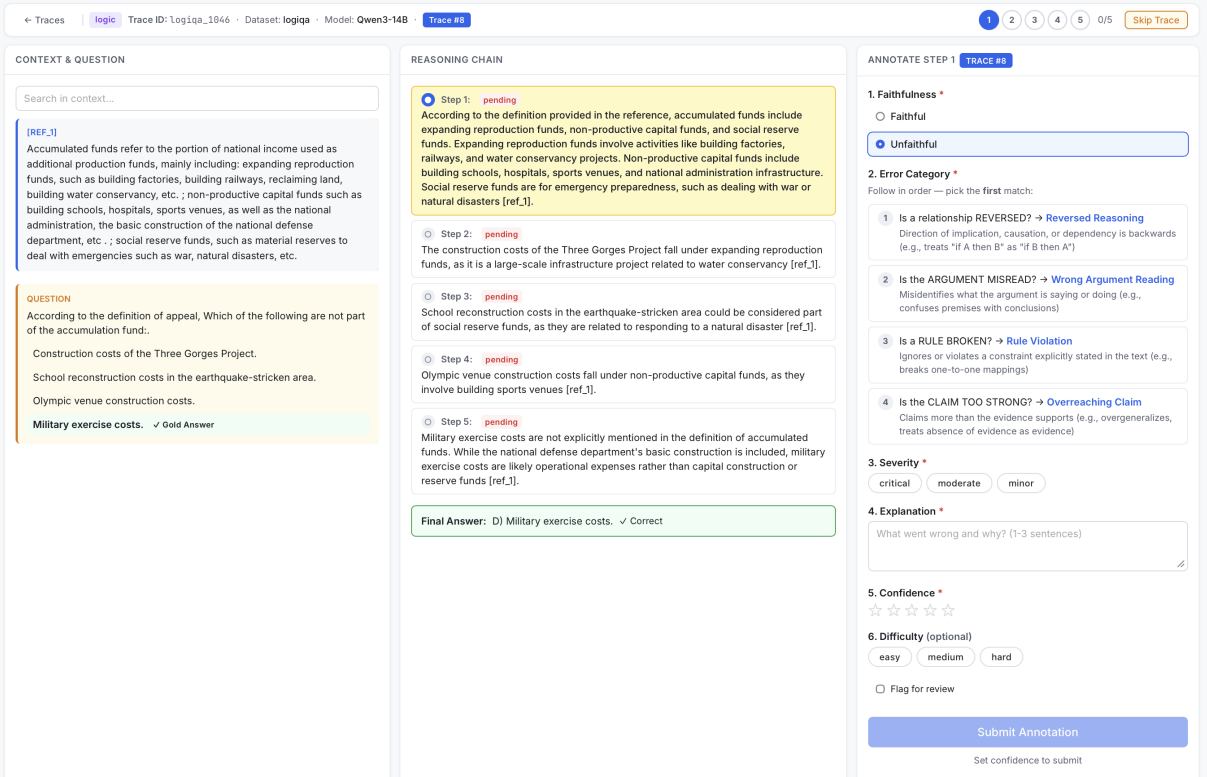

Figure 10: Annotation interface for human adjudication of GRACE test traces. The left panel shows the source context and question, the center panel lists the reasoning steps, and the right panel contains the per-step annotation form with faithfulness label, error category, severity, explanation, and confidence fields.

TRACE GENERATION

 # Task
Answer the question below by detailed reasoning step-by-step. Base your reasoning only on the provided references, then select the best option.

 # Instructions
• Each reasoning step must be grounded in specific information from the references.
• At the end of each step, cite the references you used in brackets (e.g., [ref_1, ref_2]).
• If a step is a logical deduction from previous steps rather than directly from a reference, you may omit the citation.
• Do NOT introduce any information that is not present in the references.

 # References
{references_text}

 # Question
{question}

 # Options *[Only for multiple choice questions]*
{options_text}

 # Output Format
Step 1: ⟨reasoning based on references⟩ [ref_N, . . .]
Step 2: ⟨reasoning based on references⟩ [ref_N, . . .]
. . .
Final Answer: [For free-text] ⟨answer⟩ | [For multiple choice] ⟨selected option, e.g., A) . . . ⟩

Figure 11: Prompt template for trace generation.

GROUNDEDNESS CLASSIFICATION

 # Task
Determine whether a question's correct answer can be derived SOLELY from information in the provided passage, or whether answering correctly requires introducing new information not present in the passage.

 # Definitions
A question is **GROUNDED** if:
• The correct answer is a conclusion, inference, or fact derivable from the passage alone
• Every piece of information needed to verify the answer exists in the passage
• The reasoning stays within the scope of what the passage states

A question is **NOT_GROUNDED** if:
• The correct answer introduces NEW hypothetical facts not in the passage (e.g., "which of the following, if true, would weaken the argument?")
• The question asks to identify logical FLAWS or ERRORS in reasoning (requires meta-knowledge about fallacy types, not passage content)
• The question asks for unstated ASSUMPTIONS (the answer is a premise NOT in the passage)
• The question asks what would STRENGTHEN, SUPPORT, or JUSTIFY the argument (answer is new evidence not in the passage)
• The question asks what would be useful for EVALUATING the argument (answer is new evaluative criteria)

 # Input
Passage: {context}
Question: {question}
Answer Choices: {options}
Correct Answer: {correct_answer}

 # Instructions
1. Provide your reasoning in 2–4 sentences.
2. On the LAST line, provide your classification in exactly this format:
Classification: **GROUNDED** or **NOT_GROUNDED**

Figure 12: Prompt template for groundedness classification.

OPEN CRITIQUE

You will evaluate a SINGLE reasoning step from an LLM's chain-of-thought. Determine whether this step is FAITHFUL to the provided context.

 # Definitions
• **FAITHFUL**: All claims in this step are supported by or logically derivable from the context.
• **UNFAITHFUL**: The step contains errors relative to the context (see below).
 # What counts as an error
An error is ANY way a reasoning step fails to be faithful to the context. This includes but is not limited to: Introducing information not present in the context; Contradicting something the context states; Drawing conclusions that don't logically follow from the premises; Referencing incorrect parts of the context; Missing critical information that was available; Making leaps of logic or any other form of unfaithfulness.
Do NOT limit yourself to these examples. Describe what you actually observe.

 # Input
Context: {context}
Question: {question} {options_block}
Step being evaluated (Step {step_id} of {total_steps}): "{step_text}"
Previous steps for chain context: {previous_steps}

 # Output Format
Respond in exactly this format:

<error_description>If unfaithful: 2-3 sentences. If faithful: none.</error_description>
<judgment>faithful | unfaithful</judgment>
<error_tags>1-3 comma-separated tags (e.g., fabricated entity). Or none.</error_tags>
<context_evidence>Quote from context, or none.</context_evidence>

Figure 13: Prompt template for open critique phase. The evaluator freely describes what went wrong without seeing any predefined taxonomy, producing unbiased error descriptions used for taxonomy discovery.

GRACE EVALUATION PROMPT

You are a faithfulness evaluator for the GRACE benchmark, which assesses step-level faithfulness of LLM reasoning traces. Your job is to read EVERY reasoning step, compare each against the provided context, and determine its faithfulness.

 # Task
For EACH step in the reasoning trace below, determine whether it is FAITHFUL or UNFAITHFUL based on the reference context. If a step is UNFAITHFUL, classify the error into exactly ONE category from the taxonomy.

 # Definitions
• **Faithful**: All claims in the step are supported by or logically derivable from the provided context.
• **Unfaithful**: The step contains information that conflicts with, goes beyond, or misuses the provided context.

 # Taxonomy
{taxonomy_block}

 # Context
{context}

 # Question
{question}
{options_block}

 # Reasoning Trace (total_steps steps)
{trace_steps}

 # Instructions
Evaluate ALL {total_steps} steps above. For EACH step, respond in exactly this format:

<step id="1">
<explanation>Compare the step's claims against the context. 2-4 sentences.</explanation>
<faithfulness>faithful | unfaithful</faithfulness>
<error_category>Exact category name (e.g., {category_examples}). Only if unfaithful; write "none" if faithful.</error_category>
</step>
<step id="2">
...
</step>

You MUST produce one <step> block for each of the {total_steps} steps, in order from step 1 to step {total_steps}.

Figure 14: Prompt template for GRACE step-level faithfulness evaluation. The {taxonomy_block} and {category_examples} are dynamically populated depending on whether the reasoning task belongs to the evidence or logic track.