Hop, Skip, and Overthink: Diagnosing Why Reasoning Models Fumble during Multi-Hop Analysis

Anonymous ACL submission

Abstract

The emergence of reasoning models and their integration into AI chatbots have led to breakthroughs in solving complex problems that requires multi-step thought processes. Yet, a complete understanding of their reasoning error patterns is missing. In this paper, we systematically investigate the reasoning failures of contemporary language models on multi-hop question answering tasks. We develop a novel, nuanced error analysis framework that evaluates the relevance and completeness of generated reasoning steps through four complementary dimensions: *hop precision* and *hop recall* over evidence contexts; *overthinking* that captures redundant or unnecessary reasoning; and *question misinterpretation* to reflect failures that arise prior to reasoning. Through rigorous human annotation, supported by complementary automated metrics, our exploration uncovers intricate error patterns often hidden by accuracy-centric evaluations. This investigative approach provides deeper insights into the cognitive limitations of current models and offers actionable guidance toward enhancing reasoning quality, transparency, and robustness of future language modeling efforts.

1 Introduction

Large language models (LLMs) have demonstrated remarkable performance on multi-hop question answering (QA) benchmarks, such as HotpotQA (Yang et al., 2018), where success requires sourcing knowledge from multiple documents. MuSiQue (Trivedi et al., 2022) extends this task by posing harder questions that reduce shortcut reasoning and provide explicit reasoning paths to better assess multi-step inference. The traditional evaluation metrics employed in these tasks, such as the final answer accuracy or the F1 score, fail to distinguish between genuine multi-step inference, simple memorization (as exposed by counterfactual benchmarks such as CofCA (Wu et al., 2025)), and over-reliance on dataset artifacts. Moreover, emerging studies (Sakarvadia, 2024; Agarwal et al., 2024) show that errors may stem from inadequate knowledge recall, misinterpretation of question intent, or retrieval failures in retrieval-augmented settings.

Motivated by these limitations, we move beyond answer correctness to analyze how and why reasoning models fail in multi-hop QA. We introduce a diagnostic framework that characterizes reasoning behavior along four complementary dimensions. We first define a **hop** as the basic unit of analysis, corresponding to the use of a previously unseen source document during reasoning. Reasoning quality is then quantified using **hop precision**, which measures the relevance of generated hops, and **hop recall**, which captures whether all required reasoning steps are covered. We additionally track **overthinking**, reflecting redundant or unnecessary reasoning beyond the required hops, and annotate **question misinterpretation**, capturing failures that occur prior to reasoning. Together, these dimensions support fine-grained qualitative annotation and targeted quantitative analysis, with reasoning categories derived directly from combinations of hop precision and hop recall.

Based on this framework, our contributions are as follows. We curate 1440 human annotations of failure modes across 6 reasoning models and 3 diverse datasets representing modern AI-based search process of traversing multiple documents: 2WikiMultiHopQA, HotpotQA, and MuSiQue. We quantify the distribution of reasoning errors for each model, identifying common reasoning issues, such as, mid-reasoning breakdowns, unnecessary steps, and correct answers derived from flawed logic, especially in questions with many entities or confusing information. Finally, we assess an LLM-as-a-Judge framework, which aligns well with human annotations on simpler datasets but reveals limitations on more complex ones. This supports the use of scalable, semi-automated evaluation for

reasoning analysis.

2 Related Works

Despite advances in chain-of-thought prompting, LLM explanations often diverge from true reasoning paths and can be post-hoc rationalizations (Wei et al., 2023; Bao et al., 2024). Dedicated reasoning models frequently outperform standard LLMs on medium tasks but show scaling limits and even collapse on high-complexity problems despite detailed traces (Shojaee et al., 2025). Meanwhile, standard metrics emphasize answer correctness and miss reasoning quality; multihop QA exposes shortcutting, and heuristic faithfulness measures can mask failures (Ishii et al., 2024; Lanham et al., 2023). Recent work on intermediate errors finds that correcting flawed steps, detecting process errors, and adding explicit premises can improve robustness and clarity (Li et al., 2024; Zheng et al., 2024; Mukherjee et al., 2025). Yet hallucinations in long-form outputs remain hard to detect, and repeated mistakes persist without explicit supervision (Kamoi et al., 2024; Tong et al., 2024). Beyond QA, Olympiad-math and multimodal benchmarks reveal shallow or incomplete reasoning even when answers are correct, underscoring the need for fine-grained analyses; we build on this by explicitly annotating multi-hop traces and categorizing failure patterns across diverse QA datasets for scalable evaluation (Mahdavi et al., 2025; Yan et al., 2024). To fill these gaps, we introduce a hop-based diagnostic taxonomy with meta-markers and a LLM-as-a-Judge pipeline to scale fine-grained annotations with high fidelity and inter-annotator agreement.

3 Method

3.1 Task Formalization

We view *multi-hop question answering (QA)* as a structured reasoning process that synthesizes evidence across multiple inter-dependent steps. Following prior work (Trivedi et al., 2022), we represent the reasoning process for a question as a directed acyclic graph (DAG) capturing dependencies between reasoning steps.

Formally, given a question q and a document collection $D = \{d_1, d_2, \ldots, d_m\}$, we define a reasoning graph $G_q = (V, E)$, where each node $v \in V$ corresponds to a reasoning step, and an edge $(v_j, v_i) \in E$ indicates that step v_i depends on the output of step v_j.

To ground this abstract reasoning structure in observable model behavior, we define a **context hop** ("hop") as the model's access to a **unique document context** $d_j \in D$ during inference. For simplicity, we count each context at most once within a reasoning path, regardless of how many facts it contributes. Consequently, the hop count reflects the number of distinct documents used by the model during inference to support an answer.

Let $\mathcal{P} = (h_1, h_2, \ldots, h_{N_{\text{model}}})$ denote the ordered sequence of hops induced by a model during inference, where each h_i corresponds to the model's first access to a unique document d_i, and let $\mathcal{P}^*$ denote the gold-standard reasoning path. We define the model hop count as $N_{\text{model}} = |\mathcal{P}|$ and the gold hop count as $N_{\text{gold}} = |\mathcal{P}^*|$.

3.2 Metrics

We define four metrics to quantify reasoning behavior in multi-hop question answering (QA). These metrics capture document-level coverage, relevance, efficiency, and semantic alignment of the model's reasoning process.

Hop recall. Hop recall measures the fraction of gold document hops that are recovered by the model during inference:

$$\text{Hop-Recall} = \frac{|\mathcal{P} \cap \mathcal{P}^*|}{|\mathcal{P}^*|} = \frac{|\mathcal{P} \cap \mathcal{P}^*|}{N_{\text{gold}}}.$$

Low hop recall indicates missing gold document hops and insufficient coverage during reasoning.

Hop precision. Hop precision measures the fraction of document hops retrieved by the model that correspond to gold document hops:

$$\text{Hop-Precision} = \frac{|\mathcal{P} \cap \mathcal{P}^*|}{|\mathcal{P}|} = \frac{|\mathcal{P} \cap \mathcal{P}^*|}{N_{\text{model}}}.$$

Low hop precision indicates the inclusion of irrelevant document hops, reflecting over-retrieval during reasoning.

Overthinking. Overthinking captures reasoning inefficiency rather than factual or logical error. It is annotated when the model introduces extraneous information from relevant documents that does not advance the reasoning chain, or redundantly revisits the same entities, relations, or facts. Overthinking is independent of hop recall and hop precision and may co-occur with any reasoning outcome.

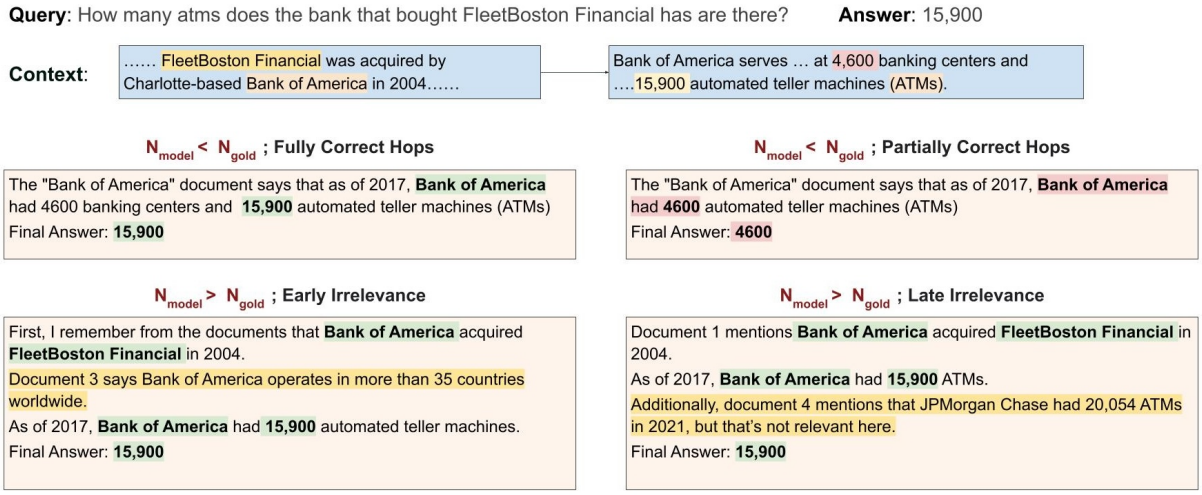

Figure 1: **Examples of Reasoning Error Categories.** Representative outputs illustrating the main error categories in multi-hop reasoning for a single example. The correct entities are highlighted in green, incorrect in red, and irrelevant or extraneous information in yellow.

Question misinterpretation. Question misinterpretation is annotated when the model incorrectly frames the question or target entities at the outset, resulting in a fundamentally misaligned reasoning chain that cannot be explained by hop recall or hop precision alone.

4 Experimental Setup

Models. We analyze six language models that span a range of architectures, parameter scales, and accessibility. Our primary focus is on four open-source distilled models, DEEPSEEK-R1-DISTILL-LLAMA-8B, DEEPSEEK-R1-DISTILL-LLAMA-70B, DEEPSEEK-R1-DISTILL-QWEN-7B, and DEEPSEEK-R1-DISTILL-QWEN-14B, which we subsequently refer to as R1-DISTILL-* for brevity. To complement these, we include two original reasoning models: CLAUDE 3.7 SONNET, a proprietary reasoning model, and DEEPSEEK-R1, an open-weight reasoning model. For all DeepSeek models, we set the generation temperature to 0.6, following the recommendations of Liu et al. (2024), to mitigate endless repetition or incoherent outputs. For Claude 3.7 Sonnet, we use a deterministic setting with the temperature set to 0.

Datasets. We evaluate reasoning across three multi-hop QA datasets of increasing difficulty: 2WikiMultiHopQA (Ho et al., 2020), which emphasizes structured multi-hop reasoning; HotpotQA (Yang et al., 2018), which includes distractors and comparisons; and MuSiQue (Trivedi et al., 2022), a high-complexity benchmark designed to minimize shortcuts through dense context and sub-question dependencies. Dataset details are provided in Appendix Table 5.

Question types. We adopt existing question-type annotations from the underlying datasets and consider five multi-hop question types based on their logical structure: *Compositional*, *Comparison*, *Intersection*, *Inference*, and *Bridge Comparison*. These categories reflect the types of reasoning steps required to arrive at the correct answer. We analyze how reasoning behavior and error patterns vary across these question types in Appendix A.3, with detailed definitions and illustrative examples provided in Appendix Table 6.

4.1 Annotation Process

Appendix Figures 4 and 5 shows the custom annotation interface, which was configured to support structured error labeling, flag toggling, and hop trace visualization by human annotators. Using this interface, we (10 human judges who are NLP experts) annotated 1,440 model outputs; after discarding examples with missing answers due to dataset artifacts, 1,080 remained for analysis. Our structured human annotation pipeline comprises two key stages:

1. **Sampling and generation.** We uniformly sampled 240 questions across HotpotQA, 2WikiMultiHopQA, and MuSiQue. Six models answered each question using a standardized prompting strategy designed to minimize instruction-induced bias.

Dataset	Model	Hop-Prec	Hop-Rec	Misint. rate	Overthinking rate	Ans Accuracy
HotpotQA	R1-DISTILL-LLAMA-8B	0.6522	0.9267	4.00%	29.3%	76%
	R1-DISTILL-LLAMA-70B	0.5867	0.9867	0.00%	18.7%	92%
	R1-DISTILL-QWEN-7B	0.5017	0.9459	1.33%	12.0%	80%
	R1-DISTILL-QWEN-14B	0.6371	0.9795	0.00%	41.3%	94.67%
	DEEPSEEK-R1	0.6167	0.9800	0.00%	22.7%	94.67%
	CLAUDE 3.7 SONNET	0.8408	1.0000	1.33%	28.0%	92%
2Wiki	R1-DISTILL-LLAMA-8B	0.8393	0.9676	1.47%	41.2%	85.29%
	R1-DISTILL-LLAMA-70B	0.9103	0.9824	0.00%	27.9%	91.18%
	R1-DISTILL-QWEN-7B	0.7089	0.9713	1.47%	19.1%	76.47%
	R1-DISTILL-QWEN-14B	0.8549	0.9860	0.00%	26.5%	95.59%
	DEEPSEEK-R1	0.9002	0.9897	0.00%	22.1%	95.59%
	CLAUDE 3.7 SONNET	0.9602	0.9896	0.00%	30.9%	94.12%
MuSiQue	R1-DISTILL-LLAMA-8B	0.8028	0.8517	16.07%	48.3%	48.21%
	R1-DISTILL-LLAMA-70B	0.7648	0.9167	0.00%	53.3%	78.57%
	R1-DISTILL-QWEN-7B	0.5394	0.8178	17.86%	41.7%	33.93%
	R1-DISTILL-QWEN-14B	0.7731	0.9111	1.79%	61.7%	75.00%
	DEEPSEEK-R1	0.7097	0.9600	0.00%	36.7%	89.29%
	CLAUDE 3.7 SONNET	0.8693	0.9833	0.00%	50.0%	80.36%

Table 1: Hop-level reasoning quality, error rates, and final answer accuracy across datasets. We report hop precision and hop recall, the rate of question misinterpretation, the overthinking rate, and overall final answer accuracy for each model and dataset.

2. **Annotation.** The final answers were evaluated for correctness using automated matching, with manual verification for paraphrased or non-exact responses. Simultaneously, we annotated: (a) N_{model}, (b) the *coverage count* $\left|P_{\text{model}} \cap P_{\text{gold}}\right|$, representing the number of gold hops retrieved by the model, (c) an *Overthinking* marker, (d) *Early/Trailing Irrelevance* if any, and (e) *Question Misinterpretation* marker. Later we compute hop-recall, and hop-precision based on those.

5 Human Evaluation Results

5.1 Hop-Level Reasoning Metrics and Answer Accuracy

Table 1 summarizes the main results across all datasets and models, reporting hop-level precision and recall, overthinking rate, and final answer accuracy. The results reveal clear differences in reasoning behavior across models and datasets, and enable a systematic analysis of how hop-level properties relate to end-to-end correctness.

Hop precision vs. final answer accuracy. We first examine whether hop precision is predictive of final answer accuracy within each dataset. Within a given dataset, hop precision exhibits a weak-to-moderate positive relationship with answer accuracy, particularly when precision is low. Models with substantially lower hop precision tend to produce less accurate final answers, indicating that hallucinated or irrelevant hops negatively affect correctness. However, increase in hop precision beyond a moderate level do not consistently translate into higher accuracy. Models with similar hop precision often differ markedly in final answer accuracy, suggesting that hop precision alone is insufficient to explain end-to-end performance once spurious hops are largely controlled.

Hop recall vs. final answer accuracy. Hop recall shows a stronger and more direct relationship with final answer accuracy within each dataset. Models that fail to retrieve all required hops rarely produce correct answers, even when hop precision is relatively high. In contrast, models with high hop recall retain the potential for correct answers, though correctness is not guaranteed. This asymmetry indicates that hop recall functions as a near-necessary condition for answer correctness: incomplete coverage of required facts is a definitive bottleneck, since missing evidence fundamentally limits the model's ability to derive the correct answer within a dataset.

Overthinking rate vs. final answer accuracy. We next analyze the relationship between overthinking behavior and final answer accuracy within datasets. Overthinking rate generally exhibits a negative association with answer correctness, though the effect is not strictly linear. Moderate levels of

overthinking do not necessarily harm performance, as sometimes additional reasoning steps may be redundant but non-disruptive. However, higher overthinking rates are associated with reduced accuracy. This effect is most evident on complex benchmarks such as MuSiQue, where overthinking rates are substantially higher across models (e.g., reaching 61.7% for DEEPSEEK-R1-DISTILL-QWEN-7B), suggesting that task complexity, rather than model scale, is the primary driver of overthinking. We also observed that most of the incorrect answers had overthinking in reasoning chain. Overthinking is not merely harmless elaboration, but a systematic driver of reasoning collapse and failure to reach a final answer.

Model size and reasoning behavior. We analyze how model size relates to hop-level reasoning behavior within each dataset. Larger models often achieve higher hop recall and lower overthinking rates, contributing to improved final answer accuracy. However, scaling does not consistently resolve deeper reasoning issues: even the largest models (CLAUDE 3.7, DEEPSEEK-R1) still exhibit precision errors and overthinking, and on MuSiQue, CLAUDE overthinks more than smaller models such as DEEPSEEK-R1-DISTILL-LLAMA-8B and DEEPSEEK-R1-DISTILL-QWEN-7B. Overall, model scale alone does not determine reasoning behavior; training and architectural choices remain critical in complex or distractor-heavy settings.

Dataset characteristics and reasoning metrics. Finally, we examine how dataset characteristics influence hop-level reasoning metrics. More challenging datasets induce higher overthinking rates and place greater demands on hop recall, particularly when longer or more compositional reasoning chains are required. In contrast, datasets with simpler reasoning structures exhibit lower overthinking rates and more stable relationships between hop-level metrics and final answer accuracy. These observations suggest that dataset difficulty and structure significantly shape model reasoning behavior, with overthinking and recall failures becoming more prevalent as reasoning complexity increases.

5.2 Reasoning Quality vs Final Accuracy

Table 2 reports reasoning quality and final answer accuracy conditioned on reasoning correctness; the trends are discussed below:

Reasoning is not sufficient. Even when reasoning is perfect, models do not always produce correct answers, particularly on MuSiQue where accuracy under perfect reasoning drops below 90% for several models. This indicates that correct hop selection alone is insufficient for reliable answer composition.

Correct answers despite wrong reasoning. Several models, notably DEEPSEEK-R1 and DEEPSEEK-R1-DISTILL-QWEN-14B, achieve high final accuracy even when their reasoning traces are incorrect. This suggests that answer correctness can arise independently of the explicit reasoning quality.

Model-dependent coupling. Claude shows the strongest coupling between reasoning quality and final accuracy, with substantial performance degradation when reasoning is incorrect. In contrast, DeepSeek-style models are more robust to reasoning errors.

Distilled models exceed DeepSeek-R1 in reasoning quality. Across both simpler and moderately complex benchmarks, larger distilled models exhibit reasoning quality that surpasses the original DEEPSEEK-R1 model. In particular, the distilled 70B and 14B variants achieve higher rates of perfect reasoning than DEEPSEEK-R1 across datasets.

Dataset dependence. Reasoning–answer alignment varies significantly by dataset, with HotpotQA exhibiting near-perfect accuracy given correct reasoning, while MuSiQue shows the weakest alignment. This indicates that dataset structure strongly influences how reliably correct reasoning leads to correct answers.

5.3 Hop Precision vs Answer Accuracy

We analyze how different forms of hop-level precision failures affect final answer accuracy, distinguishing between irrelevance introduced during reasoning versus after all required reasoning steps are completed. We focus on low-precision examples where $N_{\text{model}} > N_{\text{gold}}$ and define two error categories.

- **Early irrelevance.** The model introduces irrelevant hops before or interleaved among the gold reasoning steps in the ordered hop sequence.
- **Trailing irrelevance.** The model completes the gold reasoning hops but continues with additional irrelevant hops afterward.

Dataset	Model	(%) Perfect Reasoning	Acc@Perfect Reasoning	Acc@Imperfect Reasoning
HotpotQA	R1-DISTILL-LLAMA-8B	38.7	93.1	65.2
	R1-DISTILL-LLAMA-70B	32.0	100.0	88.2
	R1-DISTILL-QWEN-7B	26.7	90.0	76.4
	R1-DISTILL-QWEN-14B	41.3	100.0	90.9
	DEEPSEEK-R1	32.0	100.0	92.2
	CLAUDE 3.7 SONNET	62.7	100.0	78.6
2WikiMultiHopQA	R1-DISTILL-LLAMA-8B	69.1	93.6	66.7
	R1-DISTILL-LLAMA-70B	85.3	94.8	70.0
	R1-DISTILL-QWEN-7B	52.9	83.3	68.8
	R1-DISTILL-QWEN-14B	79.4	94.4	100.0
	DEEPSEEK-R1	77.9	96.2	93.3
	CLAUDE 3.7 SONNET	85.3	98.3	70.0
MuSiQue	R1-DISTILL-LLAMA-8B	33.3	70.0	40.0
	R1-DISTILL-LLAMA-70B	48.3	89.7	71.0
	R1-DISTILL-QWEN-7B	13.3	75.0	28.8
	R1-DISTILL-QWEN-14B	43.3	100.0	58.8
	DEEPSEEK-R1	38.7	89.7	84.8
	CLAUDE 3.7 SONNET	63.3	89.5	68.2

Table 2: Reasoning quality and conditional final answer accuracy across datasets. Perfect reasoning corresponds to $N_{\text{model}} = N_{\text{gold}}$ with fully correct hops, while all other responses are categorized as imperfect reasoning.

Table 3 analyzes the effect of early versus trailing irrelevance in model reasoning on final answer accuracy.

Early irrelevance is more frequent and more harmful than trailing irrelevance. Across all datasets, Early Irrelevance occurs as often or more often than trailing irrelevance, especially for smaller models (e.g., DEEPSEEK-R1-DISTILL-QWEN-7B, DEEPSEEK-R1-DISTILL-LLAMA-8B). Accuracy under early irrelevance is consistently lower than or comparable to trailing irrelevance, indicating that hallucinations before or during required hops are particularly damaging.

Trailing irrelevance is comparatively benign. Despite being common, Trailing Irrelevance often preserves high final answer accuracy (e.g., ≥ 0.85 for most large models on HotpotQA and 2Wiki). This suggests that once all required hops are completed, additional irrelevant hops rarely disrupt answer correctness

Perfect precision (no irrelevance) is rare but highly reliable. Examples with no irrelevance (i.e., perfect hop precision;) constitute a very small fraction across datasets, often near zero for HotpotQA and 2Wiki. However, when present, these cases almost always yield high accuracy, reinforcing that precision errors dominate reasoning failures.

Precision failures dominate over recall failures in practice. The table shows far more examples with irrelevance than with perfect precision but missing hops, implying that over-generation is a more prevalent failure mode than under-generation. Improving hop precision may therefore yield larger gains than improving recall alone.

5.4 Hop-wise Error Distribution

To examine how reasoning evolves across multi-hop chains, we analyze errors by hop count (Appendix Table 8.

Larger models are more stable. As the hops increase, most models exhibit a clear drop in fully correct reasoning ($N_{\text{model}} = N_{\text{gold}}$). For example, DEEPSEEK-R1-DISTILL-LLAMA-8B has 53% on 2-hop questions, but this drops to 16% for 3-hop and just 9% for 4-hop examples. In contrast, larger models such as DEEPSEEK-R1-DISTILL-LLAMA-70B and CLAUDE show much greater stability across hops. This suggests these models have a stronger capacity to follow and complete longer reasoning chains without deviation.

Overhopping dominates harder questions. For 4-hop questions, the most prominent error across several models is early irrelevance ($N_{\text{model}} > N_{\text{gold}}$). This is especially clear for DEEPSEEK-R1-DISTILL-QWEN-7B , where 73% of 4-hop examples are categorized as early irrelevance. Both CLAUDE and DEEPSEEK-R1-DISTILL-QWEN-

Dataset	Model	(%)Early Irrelevance	(%)Trailing Irrelevance	(%)No Irrelevance	Acc@Early Irrelevance	Acc@Trailing Irrelevance	Acc@No Irrelevance
HotpotQA	LLaMA 8B	25.3	26.7	2.7	0.74	0.75	0.00
	LLaMA 70B	40.0	28.0	0.0	0.87	0.90	–
	Qwen 7B	48.0	20.0	0.0	0.75	0.87	–
	Qwen 14B	41.3	17.3	0.0	0.87	1.00	–
	DeepSeek-R1	26.7	40.0	0.0	0.85	0.97	–
	Claude 3.7	20.0	14.7	0.0	0.80	0.82	–
2Wiki	LLaMA 8B	16.2	11.8	1.5	0.82	0.50	1.00
	LLaMA 70B	7.4	7.4	0.0	0.80	0.60	–
	Qwen 7B	25.0	16.2	2.9	0.59	0.91	0.50
	Qwen 14B	7.4	13.2	0.0	0.86	0.89	–
	DeepSeek-R1	2.9	19.1	0.0	1.00	0.85	–
	Claude 3.7	7.4	5.9	0.0	0.80	0.75	–
MuSiQue	LLaMA 8B	28.6	3.6	7.1	0.48	0.50	0.25
	LLaMA 70B	21.4	17.9	5.4	0.79	0.80	0.33
	Qwen 7B	37.5	14.3	7.1	0.34	0.63	0.25
	Qwen 14B	26.8	17.9	10.7	0.75	0.90	0.50
	DeepSeek-R1	26.8	26.8	3.6	0.79	0.87	0.00
	Claude 3.7	25.0	8.9	3.6	0.80	0.80	0.50

Table 3: Distribution of irrelevance patterns and conditional final answer accuracy across datasets. Early and trailing irrelevance correspond to cases with $\mathbf{N}_{\text{model}} > \mathbf{N}_{\text{gold}}$, where irrelevant reasoning appears before or after the required hops, respectively. No irrelevance corresponds to $\mathbf{N}_{\text{model}} < \mathbf{N}_{\text{gold}}$ with fully correct hops. We report the percentage of examples and final answer accuracy conditioned on each case.

14B show 45% early irrelevance at 4 hops. These results indicate that, in more complex tasks, models frequently continue reasoning beyond what is necessary and retrieve irrelevant information.

5.5 Reasoning Behaviour by Question Type

Model behavior varies strongly with question structure (detailed in Section A.3). Across question types, we observe the following systematic patterns:

- **Bridge comparison questions** are consistently solved across models and are often resolved using surface-level entity cues aligned with pretraining statistics.
- **Symmetric comparison questions** frequently induce redundant exploration, leading to early or trailing irrelevance even when a single reasoning path would suffice.
- **Compositional questions** expose limitations in integrating disjoint facts, with smaller models often producing partially correct chains and larger models mitigating, but not fully eliminating, these failures.
- **Inference questions** are the most error-prone, lacking strong lexical anchors and exhibiting the highest rates of misinterpretation and overthinking.

6 Automated Evaluation Results

While extensive manual evaluations provide detailed and reliable insights into model reasoning behaviors, it is difficult to scale, particularly for complex queries from datasets like MuSiQue, where each annotation can take approximately four minutes per data point. Hence, We develop a framework for automating the annotation process to significantly improve evaluation efficiency.

6.1 Evaluation Workflow

We employed an LLM-as-a-Judge framework (prompts included in Appendix C.1) to automate the annotation task, using `gpt-4.1-mini`[1] as our judging model. Utilizing LLM as a judge for annotating reasoning failures helped us scale the process significantly and reduced evaluation time, achieving approximately a **20x** increase in efficiency compared to manual annotation. To ensure parity with manual annotation process and high fidelity analysis, we provided the Judge LLM with the same detailed annotation guidelines used by human annotators, but prompt-engineered the guidelines with explicit formatting instructions and clear definitions. The Judge LLM had access to 'question', 'relevant context documents', and the 'final response' from the reasoning models.

[1] A state-of-the-art model that is different from the models analyzed in this work.

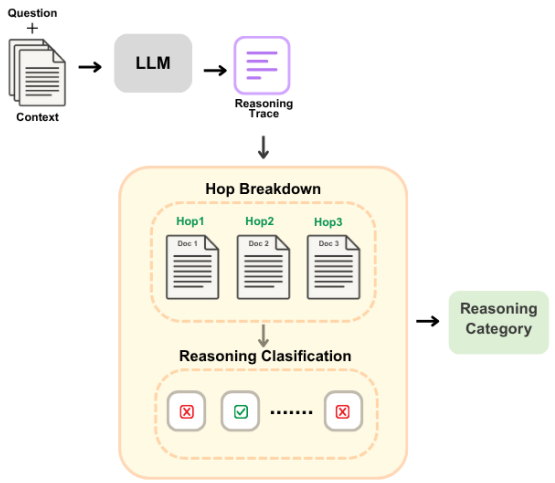

Figure 2: **Two-Step LLM-Assisted Evaluation Workflow.** A high-level overview of the two-step decomposition that improves annotation accuracy and consistency for complex multi-hop reasoning.

Consistent with findings for multi-step judging process in (Wang et al., 2023; Zheng et al., 2024), we adopt a two-step annotation process as illustrated in Figure 2. 1) **Hop breakdown**, where the Judge LLM identifies and annotates the reasoning hops present in the model's response; and (2) **Reasoning classification**, where the Judge uses these annotated hops to categorize the response into one of our predefined error categories. This decomposition significantly improved annotation accuracy and consistency, aligning with findings from recent literature indicating that multi-step judging processes enhance reliability and accuracy in complex evaluation tasks (Wang et al., 2022).

6.2 Model-Wise Agreement with Human Annotations

To further validate our LLM-as-a-Judge pipeline, we evaluate the consistency of annotations across six models and three datasets: MuSiQue, 2Wiki, and HotpotQA. Based on the results presented in Table 4, our LLM-as-a-Judge framework demonstrates promising potential for automating error categorization tasks traditionally performed by human annotators. Across all models and datasets, agreement rates vary, indicating model-specific and dataset-specific challenges. For example, models such as DeepSeek-R1 and DeepSeek-R1-Distill-LLaMA 70B exhibit notably higher agreement rates, particularly on simpler datasets like 2Wiki, achieving above 90%. Conversely, results on the more challenging MuSiQue dataset consistently shows lower agreement scores, underscoring inherent complexities and subtle reasoning errors that the Judge LLM struggles to replicate accurately.

Model	HotpotQA	2Wiki	MuSiQue
LLaMA 8B	65.3	73.5	53.3
DeepSeek-R1	76.0	91.1	62.6
LLaMA 70B	72.0	92.6	75.0
Qwen 7B	66.6	75.0	46.6
Claude 3.7	73.3	91.1	76.6
Qwen 14B	65.3	88.2	78.3

Table 4: LLM-as-a-Judge Agreement (%) with Human Annotations Across Models and Datasets. We find that across models DeepSeek-R1-Distill-LLaMA 70B and Claude has highest agreement scores.

These results imply that while LLM-as-a-Judge systems are highly effective at automating error categorization for straightforward multi-hop reasoning tasks, complexities in certain datasets highlight the continuing necessity for human judgment or more refined instructions for the Judge LLM. The observed variability underscores that further investigation is essential to understand and mitigate factors contributing to lower Judge-model agreement rates, such as nuanced reasoning steps or subtle misinterpretations. Nonetheless, the substantial reduction in annotation time and generally high fidelity in simpler contexts strongly support the viability and efficiency of integrating LLM-as-a-Judge frameworks into broader NLP evaluation pipelines.

7 Conclusion

We investigate the reasoning error patterns in six contemporary language models across three datasets spanning varying levels of multi-hop complexity. We present extensive observations that these models still suffer from low hop-precision, low hop-recall, overthinking, and question misunderstanding mistakes in complex and distractor-rich tasks. Our two-step LLM-as-a-Judge evaluation achieves up to 92% agreement with human annotations on certain datasets while reducing evaluation time by up to 20×. Our findings highlight the need for evaluation and training strategies that move beyond answer correctness toward reasoning that is both efficient and reliable, enabling more trustworthy multi-hop QA systems.

Limitations

We have focused on off-the-shelf multi-hop reasoning question answering tasks where the input involves multiple documents. We have analyzed model behaviors in terms of hopping to each evidence document. It will be interesting to explore more fine-grained definitions of hops. Future work also includes more reasoning tasks such as math, coding, planning, and so forth. It is also beyond the scope of our paper to study multi-modality models.

AI Usage

We used ChatGPT for polishing the paper writing.

References

Chirag Agarwal, Sree Harsha Tanneru, and Himabindu Lakkaraju. 2024. Faithfulness vs. plausibility: On the (un)reliability of explanations from large language models. *Preprint*, arXiv:2402.04614.

Guangsheng Bao, Hongbo Zhang, Cunxiang Wang, Linyi Yang, and Yue Zhang. 2024. How likely do llms with cot mimic human reasoning? *Preprint*, arXiv:2402.16048.

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.

Ai Ishii, Naoya Inoue, Hisami Suzuki, and Satoshi Sekine. 2024. Analysis of LLM's "spurious" correct answers using evidence information of multi-hop QA datasets. In *Proceedings of the 1st Workshop on Knowledge Graphs and Large Language Models (KaLLM 2024)*, pages 24–34, Bangkok, Thailand. Association for Computational Linguistics.

Ryo Kamoi, Sarkar Snigdha Sarathi Das, Renze Lou, Jihyun Janice Ahn, Yilun Zhao, Xiaoxin Lu, Nan Zhang, Yusen Zhang, Ranran Haoran Zhang, Sujeeth Reddy Vummanthala, Salika Dave, Shaobo Qin, Arman Cohan, Wenpeng Yin, and Rui Zhang. 2024. Evaluating llms at detecting errors in llm responses. *Preprint*, arXiv:2404.03602.

Tamera Lanham, Anna Chen, Ansh Radhakrishnan, Benoit Steiner, Carson Denison, Danny Hernandez, Dustin Li, Esin Durmus, Evan Hubinger, Jackson Kernion, Kamilė Lukošiūtė, 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. *Preprint*, arXiv:2307.13702.

Xiaoyuan Li, Wenjie Wang, Moxin Li, Junrong Guo, Yang Zhang, and Fuli Feng. 2024. Evaluating mathematical reasoning of large language models: A focus on error identification and correction. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 11316–11360, Bangkok, Thailand. Association for Computational Linguistics.

Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, and 1 others. 2024. Deepseek-v3 technical report. *arXiv preprint arXiv:2412.19437*.

Hamed Mahdavi, Alireza Hashemi, Majid Daliri, Pegah Mohammadipour, Alireza Farhadi, Samira Malek, Yekta Yazdanifard, Amir Khasahmadi, and Vasant Honavar. 2025. Brains vs. bytes: Evaluating llm proficiency in olympiad mathematics. *Preprint*, arXiv:2504.01995.

Sagnik Mukherjee, Abhinav Chinta, Takyoung Kim, Tarun Anoop Sharma, and Dilek Hakkani-Tür. 2025. Premise-augmented reasoning chains improve error identification in math reasoning with llms. *Preprint*, arXiv:2502.02362.

Mansi Sakarvadia. 2024. Towards interpreting language models: A case study in multi-hop reasoning. *Preprint*, arXiv:2411.05037.

Parshin Shojaee, Iman Mirzadeh, Keivan Alizadeh, Maxwell Horton, Samy Bengio, and Mehrdad Farajtabar. 2025. The illusion of thinking: Understanding the strengths and limitations of reasoning models via the lens of problem complexity. *Preprint*, arXiv:2506.06941.

Yongqi Tong, Dawei Li, Sizhe Wang, Yujia Wang, Fei Teng, and Jingbo Shang. 2024. Can llms learn from previous mistakes? investigating llms' errors to boost for reasoning. *Preprint*, arXiv:2403.20046.

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*, 10:533–550.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. *Preprint*, arXiv:2203.11171.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. 2023. Chain-of-thought prompting elicits reasoning in large language models. *Preprint*, arXiv:2201.11903.

Jian Wu, Linyi Yang, Zhen Wang, Manabu Okumura, and Yue Zhang. 2025. CofCA: A STEP-WISE counterfactual multi-hop QA benchmark. In *The Thirteenth International Conference on Learning Representations*.

Yibo Yan, Shen Wang, Jiahao Huo, Hang Li, Boyan Li, Jiamin Su, Xiong Gao, Yi-Fan Zhang, Tianlong Xu, Zhendong Chu, Aoxiao Zhong, Kun Wang, Hui Xiong, Philip S. Yu, Xuming Hu, and Qingsong Wen. 2024. Errorradar: Benchmarking complex mathematical reasoning of multimodal large language models via error detection. *Preprint*, arXiv:2410.04509.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.

Chujie Zheng, Zhenru Zhang, Beichen Zhang, Runji Lin, Keming Lu, Bowen Yu, Dayiheng Liu, Jingren Zhou, and Junyang Lin. 2024. Processbench: Identifying process errors in mathematical reasoning. *Preprint*, arXiv:2412.06559.

A Reasoning Taxonomy and Evaluation results

A.1 Refining Reasoning Categories

To diagnose reasoning failures in multi-hop QA, we refined our error taxonomy through three iterative stages. Each stage addressed prior shortcomings and improved inter-annotator agreement, as shown in Appendix Figure 3. Full definitions for Stage 1 and Stage 2 are in the next section. Stage 3 denotes the final annotation strategy mentioned in the main paper.

Stage 1: Coarse Conceptual Labels Our initial taxonomy used four loosely defined labels: *Effective*, *Underthinking*, *Overthinking*, and *Faulty*. These arose from manual trace inspection but lacked clear definitions. Annotators struggled to distinguish between concise reasoning and underthinking, or between verbose, incorrect reasoning and overthinking. The lack of a formal notion of reasoning hops made error tracing difficult. *Faulty* served as a catch-all for various errors, reducing analytical usefulness.

Stage 2: Structured Hop-Based Categorization In the second stage, we introduced a 10-category taxonomy based on N_{model}, N_{gold}, hop correctness, and answer accuracy to support structured error analysis. As manual evaluation scaled, new ambiguities emerged. Category 8 (early hallucinations) often overlapped with Category 6 (underspecified chains) and question misinterpretation. Annotators also struggled to distinguish shortcut reasoning from flawed logic. Thus, this structurally driven categories needed stronger semantic clarity.

Stage 3: Final Schema with Meta-Evaluation Markers The final schema addressed these issues through clearer definitions. We formally defined a reasoning **hop**, excluding repeated entity mentions within the same document to avoid inflated hop counts. We also distinguished overthinking via cross-document exploration ($N_{\text{model}} > N_{\text{gold}}$) from that in verbose or circular reasoning within a document, using a separate overthinking flag.

Stage 1 categories (coarse taxonomy)

1. **Effective reasoning:** The model performs all required reasoning steps and correctly answers the question. The explanation is concise, coherent, and logically complete.

2. **Underthinking:** The model provides insufficient reasoning, skipping essential steps or offering vague justifications. The response may appear shallow or overly brief, regardless of answer correctness.

3. **Overthinking:** The model introduces excessive or tangential reasoning, often by exploring irrelevant paths or repeating information. This may include unnecessary document traversal or redundant entity comparisons.

4. **Faulty reasoning:** The reasoning chain is logically flawed or factually incorrect. This may involve wrong inference, unsupported claims, or internal contradictions, even if the structure appears complete.

Stage 2 categories (structured taxonomy) Let N_{model} denote the number of reasoning steps predicted by the model, and N_{gold} denote the number of hops required according to the gold standard.

1. **Category 1: $N_{\text{model}} = N_{\text{gold}}$; all hops correct; final answer correct.** The model follows the required inference path, makes all correct hops and provides the correct final answer.

2. **Category 2: $N_{\text{model}} = N_{\text{gold}}$; all hops correct; final answer incorrect.** The reasoning

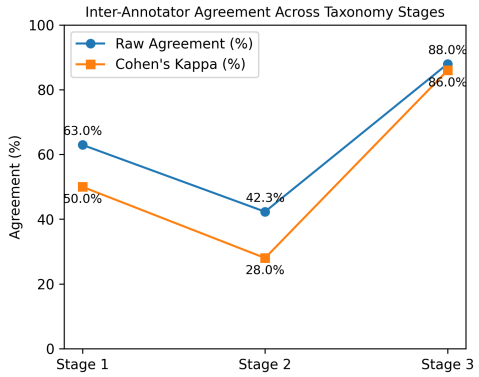

Figure 3: **Improvement in Inter-Annotator Agreement Across Refinement Stages.** Raw agreement and Cohen's kappa both increase substantially as the reasoning error taxonomy evolves from loosely defined to formally structured categories, with the highest agreement achieved after Stage 3 refinements.

path is structurally correct, but the final answer is wrong due to errors in aggregation or conclusion.

3. **Category 3: $N_{\text{model}} = N_{\text{gold}}$; one or more hops incorrect/hallucinated.** The hop count matches, but one or more steps are logically or factually incorrect. The final answer may or may not be correct.

4. **Category 4: $N_{\text{model}} < N_{\text{gold}}$; all predicted hops correct; final answer incorrect.** The model correctly predicts a subset of the required hops but misses key steps, leading to an incorrect answer.

5. **Category 5: $N_{\text{model}} < N_{\text{gold}}$; all predicted hops correct; final answer correct (shortcut).** The model answers correctly using a valid but incomplete subset of required reasoning hops. A shortcut was taken.

6. **Category 6: $N_{\text{model}} < N_{\text{gold}}$; one or more hops incorrect/hallucinated.** The model generates fewer hops than required, with some being inaccurate or irrelevant. The chain is both incomplete and partially flawed.

7. **Category 7: $N_{\text{model}} > N_{\text{gold}}$; irrelevant hops after gold path (trailing overthinking).** After attempting the required reasoning, the model continues with superfluous or irrelevant steps, leading to over-generation.

8. **Category 8: $N_{\text{model}} > N_{\text{gold}}$; irrelevant or hallucinated hops before/interleaved.** Irrelevant hops occur early in the reasoning process or are interleaved with required steps, disrupting logical progression.

9. **Category 9: $N_{\text{model}} = 0$.** No reasoning path is shown; the model outputs an answer directly without generating any hops.

10. **Category 10: Question misinterpretation.** The reasoning chain reflects a misunderstanding of the question, regardless of hop count or structural form.

A.2 Dataset Details

- **2WikiMultiHopQA** (Ho et al., 2020): Includes 10 Wikipedia paragraphs per question, retrieved from structured and unstructured sources. Each instance provides gold reasoning paths and supporting facts to ensure multi-hop inference.

- **HotpotQA** (Yang et al., 2018): Each question is paired with two gold documents and eight distractors (10 in total). Designed to test both answer accuracy and reasoning transparency, including bridge and comparison questions with annotated supporting sentences.

- **MuSiQue** (Trivedi et al., 2022): Presents 20-document contexts per question. Constructed to reduce shortcut-based reasoning by enforcing sub-question dependencies and including challenging unanswerable distractors.

A.3 Question Types

The definitions and illustrative examples of all question types are provided in Appendix Table 6.

Appendix Table 7 shows reasoning error distributions across question categories for all models. Several clear trends emerge:

Bridge Comparison Questions Are Consistently Solved, Especially from 2Wiki: Bridge questions (mainly from 2Wiki) yield 94–100% fully correct hops across all models. Even smaller models like DEEPSEEK-R1-DISTILL-QWEN-7B and DEEPSEEK-R1-DISTILL-LLAMA-8B perform well, while CLAUDE and DEEPSEEK-R1-DISTILL-QWEN-14B make no errors. These questions often contain explicit entity references or co-occurrence cues resembling model pre-training patterns, enabling resolution through surface-level recognition rather than deep reasoning.

Dataset	Question Types	#Hops	Difficulty
HotpotQA	Composition, Comparison	2	Easy–Medium
2WikiMultiHopQA	Composition, Comparison, Inference, Bridge Comparison	2, 4	Medium–Hard
MuSiQue	Composition, Inference	2, 3, 4	Hard

Table 5: Comparison of datasets in terms of question types, hop complexity, and difficulty level.

Question type	Bridge entity / reasoning	Example
Compositional Requires chaining intermediate entities	Bridge: Versus	**Doc 1:** Versus (Versace) is the diffusion line of Italian..., a gift by the founder Gianni Versace. **Doc 2:** Gianni Versace was shot and killed outside... **Question:** Why did the founder of Versus die?
Inference Demands implicit reasoning via unstated bridge facts	Bridge: Grandchild	**Doc 1:** Dambar Shah was the father of Krishna Shah... **Doc 2:** Krishna Shah was the father of Rudra Shah... **Question:** Who is the grandchild of Dambar Shah?
Comparison Involves comparing attributes across entities	Compare: Age of persons	**Doc 1:** Theodor Haecker (1879–1945) was a... **Doc 2:** Harry Vaughan Watkins (1875–1945) was a... **Question:** Who lived longer, Theodor Haecker or Harry Vaughan Watkins?
Bridge comparison Combines inference followed by comparison	Bridge: Directors' Nationality	**Doc 1:** FAQ: Frequently Asked Questions directed by Carlos Atanes... **Doc 2:** The Big Money directed by John Paddy Carstairs... **Doc 3:** Carlos Atanes is a Spanish film director. **Doc 4:** John Paddy Carstairs was a British film director. **Question:** Are both directors of FAQ: Frequently Asked Questions and The Big Money from the same country?

Table 6: Examples of different question types with highlighted entities

Symmetric Structures Trigger Redundant Reasoning and Overhopping: Found in HotpotQA and 2Wiki, comparison questions show 50–68% fully correct rates, with 25–45% of errors due to early or trailing irrelevance. Their symmetric phrasing encourages exploration of both options, even when one suffices. CLAUDE occasionally bypasses intermediate hops while still producing correct answers, suggesting reliance on shortcut-style or selective reasoning paths.

Compositional Reasoning Exposes Integration Failures: Compositional questions strain models' ability to synthesize disjoint facts. Smaller models (DEEPSEEK-R1-DISTILL-QWEN-7B, DEEPSEEK-R1-DISTILL-LLAMA-8B, DEEPSEEK-R1) show many partially correct chains, even with correct hop counts. CLAUDE and DEEPSEEK-R1-DISTILL-LLAMA-70B perform better, suggesting that scale and architecture improve integration.

Inference Questions Are the Most Error-Prone and Trigger Overthinking: Inference questions, heavily present in MuSiQue and 2Wiki, demand implicit reasoning and multi-step logic without strong lexical cues. These questions yield the broadest error types, early/trailing irrelevance and misinterpretation. DEEPSEEK-R1-DISTILL-QWEN-7B answers only 10% correctly, with 30% misinterpretation. Even DEEPSEEK-R1 shows 37% trailing irrelevance. Only CLAUDEand DEEPSEEK-R1-DISTILL-LLAMA-70B manage modest control (50–55% correct), highlighting the inherent difficulty of inference.

A.4 Hops

Table 8 reports error distributions conditioned on the number of gold reasoning hops, showing a consistent degradation in fully correct reasoning and an increase in irrelevance and misinterpretation as hop complexity increases.

Model	Question Type	$N_{\text{model}} = N_{\text{gold}}$ (Fully Correct)	Early Irrelevance	Trailing Irrelevance	Question Misinterp.
R1-DISTILL-LLAMA-8B	Bridge Comparison	94	0	6	0
	Comparison	54	29	14	0
	Compositional	45	24	17	6
	Inference	15	30	15	25
R1-DISTILL-LLAMA-70B	Bridge Comparison	94	0	6	0
	Comparison	54	32	14	0
	Compositional	50	24	23	3
	Inference	55	30	5	5
R1-DISTILL-QWEN-7B	Bridge Comparison	69	19	0	12
	Comparison	32	43	21	0
	Compositional	30	37	20	5
	Inference	10	40	5	30
R1-DISTILL-QWEN-14B	Bridge Comparison	100	0	0	0
	Comparison	54	18	29	0
	Compositional	54	28	15	3
	Inference	25	40	15	15
DEEPSEEK-R1	Bridge Comparison	81	12	6	0
	Comparison	46	29	25	0
	Compositional	51	15	32	0
	Inference	21	32	37	5
CLAUDE 3.7 SONNET	Bridge Comparison	100	0	0	0
	Comparison	68	18	11	0
	Compositional	71	15	12	0
	Inference	50	40	10	0

Table 7: Error category distribution across question types. We report the percentage of examples with fully correct reasoning ($N_{\text{model}} = N_{\text{gold}}$), early irrelevance, trailing irrelevance, and question misinterpretation for each model and question type. Percentages may not sum to 100% as other error categories are omitted for clarity.

Model	Gold Hops	$N_{\text{model}} = N_{\text{gold}}$ (Fully Correct)	Early Irrelevance	Trailing Irrelevance	Question Misinterp.
R1-DISTILL-LLAMA-8B	2-hop	53	27	7	7
	3-hop	16	37	26	26
	4-hop	9	18	9	18
R1-DISTILL-LLAMA-70B	2-hop	47	20	23	23
	3-hop	47	26	21	21
	4-hop	55	18	9	9
R1-DISTILL-QWEN-7B	2-hop	27	33	17	17
	3-hop	0	21	16	32
	4-hop	0	73	9	18
R1-DISTILL-QWEN-14B	2-hop	50	23	20	7
	3-hop	42	21	21	5
	4-hop	27	45	0	27
DEEPSEEK-R1	2-hop	41	19	35	5
	3-hop	35	45	20	0
	4-hop	39	39	11	11
CLAUDE 3.7 SONNET	2-hop	67	17	13	3
	3-hop	68	21	11	0
	4-hop	45	45	0	9

Table 8: Error category distribution conditioned on the number of gold hops. We report the percentage of examples with fully correct reasoning ($N_{\text{model}} = N_{\text{gold}}$), early irrelevance, trailing irrelevance, and question misinterpretation. Percentages need not sum to 100% as other error categories are omitted for clarity.

B Additional Figures and Tables

Figures 4 and 5 illustrate the annotation interface used by human evaluators, showing the inputs provided (model reasoning traces, supporting evidence, and ground-truth answers) and the corresponding outputs, including evaluation flags and assignment to hop-based error categories.

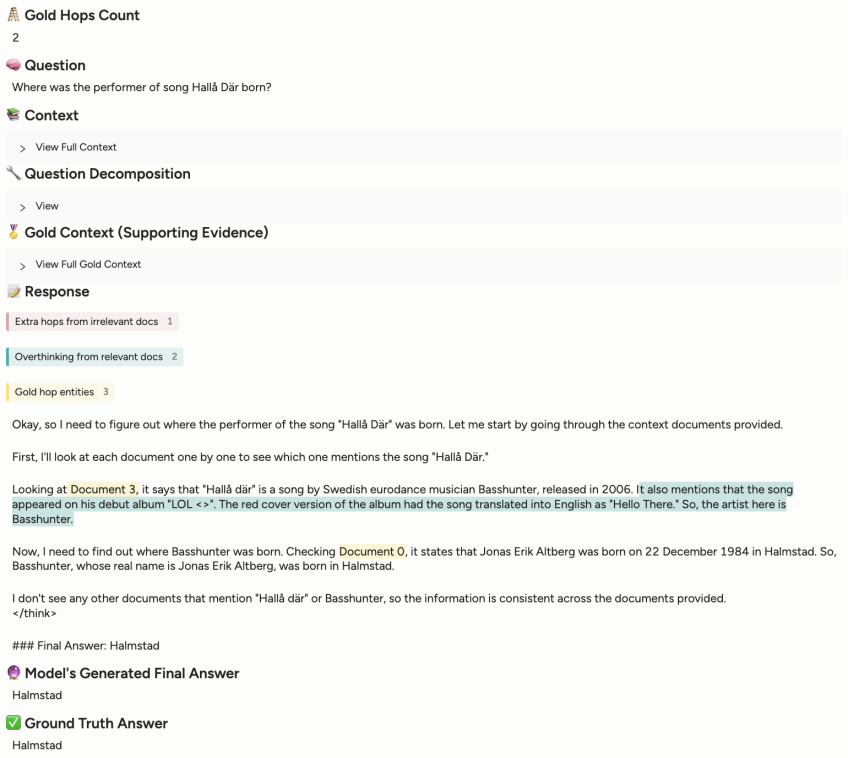

Figure 4: Example of input provided to human annotators, including the model's reasoning trace, supporting evidence, and ground-truth answer. Annotators used this interface to inspect reasoning steps and identify overthinking or irrelevant hops.

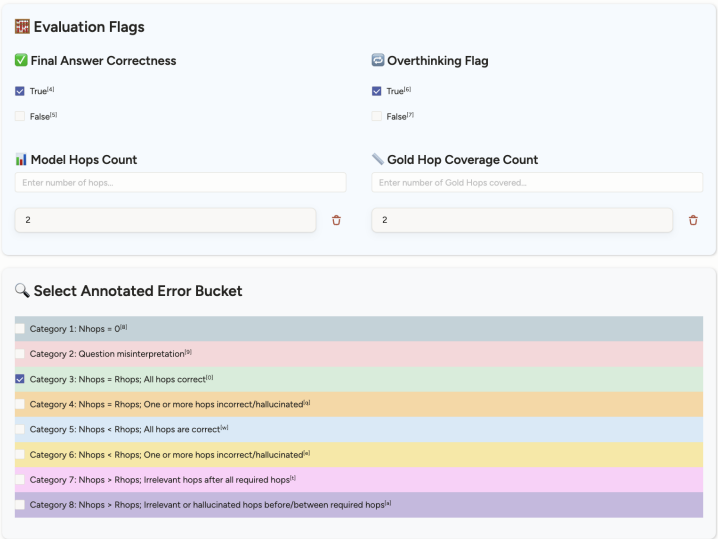

Figure 5: Example of output labeled by human annotators. The interface captures evaluation flags and assigns each example to a specific error bucket based on hop counts and reasoning behavior.

C LLM-As-A-Judge Prompts

C.1 LLM-as-a-Judge Prompting Workflow

We designed a two-stage prompting workflow for automated evaluation, aligning with the decomposition shown in Figure 2. Stage 1 extracts the reasoning hops from model responses, while Stage 2 performs classification into error categories based on hop structure and answer correctness.

Stage 1: Hop Breakdown. The Judge LLM identifies the number of reasoning hops (Nhops) taken in the model's response.

Stage 1 Prompt

```
You are a meticulous evaluator of
    multi-hop reasoning.
Your task is to identify and
    count the reasoning hops
    (Nhops)
in the model's response. Hop is a
    distinct piece of information
    (entity)
or set of sentences retrieved
    from a document. All
    information extracted
from the same document counts as
    a single hop.

### Rules for counting hops:
- Count 1 hop when the model
    extracts a fact or entity
  from a new document.
- Multiple facts from the same
    document = 1 hop.
- Each unique document mentioned
    (even if discarded later)
  counts as one hop.
- Ignore final conclusions or
    question clarifications.
- Ignore comparisons,
    calculations, or rankings.
- Ignore Re-checking or
    reiterating previously
    retrieved facts.

Input:
{
  "Question": "<question>",
  "Gold Supporting Facts": ["Doc
      1","Doc 2"],
  "Rhops_Count": <integer>,
  "Context": "<full concatenated
      context docs>",
  "Model Response": "<model's
      reasoning + answer>"
}

Output:
{
  "Nhops_Count": <integer>,
  "Hop_Breakdown": "hop 1:
      ...\nhop 2: ...",
  "Documents_Covered_in_Nhops":
      ["Doc 1","Doc 2","Doc 3"]
}
```

Stage 2: Reasoning Classification. Using the extracted hop structure, the Judge LLM compares Nhops against Rhops, evaluates final answer correctness, and assigns one predefined error category.

Stage 2 Prompt

```
You are a strict classifier of
    reasoning quality in multi-hop
    QA.
Your task is to compare Nhops
    with Rhops, check answer
    correctness,
and assign one error category.

### Steps:

1. Verify if the final answer
    matches the gold answer.
2. Conclude if Nhops=Rhops ,
    Nhops>Rhops or Nhops<Rhops?
   **VERY IMPORTANT STEP in
       DECIDING ERROR CATEGORIES**
3. Apply an Overthinking_Flag if
    redundant confirmations appear.
4. Assign exactly one error
    category from the predefined
    taxonomy.

### Classification Categories:

<List of all error types with
    definition>

Input:
{
  "Question": "<string>",
  "Gold Supporting Facts": ["Doc
      1","Doc 2"],
  "Rhops_Count": 2,
  "Gold Context": "<only gold
      docs full text>",
  "Model Response": "<model's
      reasoning + answer>",
  "Nhops_Count": <integer>,
  "Documents_Covered_in_Nhops":
      ["Doc 1","Doc 2","Doc 3"],
  "Gold Final Answer": "<string>"
}

Output:
{
  "Final_Answer_Correctness":
      true,
  "Overthinking_Flag": false,
  "Error_Category_Justification":
      "Step-by-step reasoning ...",
  "Error_Category": "Category # -
      Category Name"
}
```