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

Anonymous Author(s)
Affiliation
Address
email

Abstract

The emergence of reasoning models and their integration into practical AI chat bots has led to breakthroughs in solving advanced math, deep search, and extractive question answering problems that requires a complex and multi-step thought process. Yet, a complete understanding of why these models hallucinate more than general purpose language models is missing. In this investigative study, we systematically explore reasoning failures of contemporary language models on multi-hop question answering tasks. We introduce a novel, nuanced error categorization framework that examines failures across three critical dimensions: the diversity and uniqueness of source documents involved ("hops"), completeness in capturing relevant information ("coverage"), and cognitive inefficiency ("overthinking"). 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 fidelity, transparency, and robustness in future language modeling efforts.

1 Introduction

Language models (LMs) have demonstrated remarkable performance on multi-hop question answering (QA) benchmarks, such as HotpotQA [1], where success requires sourcing knowledge from multiple documents.MuSiQue [2] extends this task by posing harder questions that reduces shortcut reasoning and provides 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; [3], and over-reliance on dataset artifacts. Moreover, emerging studies [4, 5] show that errors may stem from missing knowledge recall, misinterpretation of question intent, or retrieval failures in retrieval-augmented settings.

With these limitations in mind, we move beyond answer correctness and undertake an investigative exploration of reasoning failures in multi-hop QA to answer a central question: How and why do reasoning models break down when stitching together information across multiple sources? To address this, we introduce a diagnostic framework that decomposes reasoning behavior along three core dimensions:

(1) **Hops** A hop is a discrete step or transition in the reasoning process where the model moves from one piece of information (e.g., a fact, source, or knowledge base entry) to another in order to bridge connections and form a complete answer. (2) **Coverage** evaluates whether all necessary reasoning steps are covered; and (3) **Overthinking** refers to whether the model meanders into unnecessary or

Submitted to 39th Conference on Neural Information Processing Systems (NeurIPS 2025). Do not distribute.

off-track reasoning. These dimensions support both qualitative annotation and targeted quantitative evaluation of reasoning fidelity.

Contributions: We comprehensively evaluate multi-hop reasoning process by introducing a structured taxonomy of 7 fine-grained error categories, and curating 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. Using the error taxonomy, we quantify the distribution of reasoning errors across models. Our study reveals common reasoning issues, such as breaking down in the middle of reasoning, adding unnecessary steps in complex cases, and providing correct answers despite flawed reasoning, especially on questions with many entities or confusing information. Finally, we evaluate the effectiveness of an LLM-as-a-Judge framework, which shows strong agreement with human annotations on simpler datasets while highlighting key 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 [6, 7]. Dedicated reasoning models frequently outperform standard LLMs on medium tasks but show scaling limits—and even collapse—on high-complexity problems despite detailed traces [8]. Meanwhile, standard metrics emphasize answer correctness and miss reasoning quality; multihop QA exposes shortcutting, and heuristic faithfulness measures can mask failures [9, 10]. Recent work on intermediate errors finds that correcting flawed steps, detecting process errors, and adding explicit premises can improve robustness and clarity [11–13]. Yet hallucinations in long-form outputs remain hard to detect, and repeated mistakes persist without explicit supervision [14, 15]. 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 [16, 17].

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 define **multi-hop QA** as the task of responding to complex questions, undertaken by reasoning models, that necessitate synthesizing information from multiple sources through a chain of reasoning steps. A **hop**, denoted by h_i, refers to a distinct reasoning step wherein the model extracts supporting evidence from a **unique document** $d_j \in D$. The number of hops in a reasoning path corresponds to the number of unique documents accessed, regardless of how much content is extracted from each.

For a question Q and a collection of m documents $D = \{d_1, d_2, \ldots, d_m\}$, the task is to predict (1) an answer A (a textual span within one of the documents in D), and (2) a reasoning path $\mathcal{P} = (h_1, h_2, \ldots, h_{n_{\text{model}}})$ representing the sequence of reasoning hops. Here, $|\mathcal{P}|$ denotes the length of the model's hop sequence, and $\mathcal{P}^*$ denotes the gold-standard reasoning path required to answer the question.

The **model hop count** is defined as $N_{\text{model}} = |\mathcal{P}|$, and the **gold hop count** is defined as $N_{\text{gold}} = |\mathcal{P}^*|$.

3.2 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 Figure 4. Full definitions for Stage 1 and Stage 2 are in the Appendix.

Stage 1: Coarse Conceptual Labels Our initial taxonomy used four loosely defined labels: *Effective*, *Underthinking*, *Overthinking*, and *Faulty*. These arose from manual trace inspection

Table 1: Definitions of reasoning categories in multi-hop QA. N_{model} denotes the number of reasoning hops executed by the model; N_{gold} is the number of required gold hops.

Reasoning category	**Definition**
$\boldsymbol{N}_{\text{model}} = \boldsymbol{N}_{\text{gold}}$; *Fully correct hops*	The model executes the exact number of required gold reasoning hops, and each hop is logically sound, complete, and correct.
$\boldsymbol{N}_{\text{model}} = \boldsymbol{N}_{\text{gold}}$; *Partially correct hops*	The model executes the correct number of reasoning steps, but one or more hops involve incorrect documents, entities, or relations. The model reasoning is partially misaligned with the gold reasoning path.
$\boldsymbol{N}_{\text{model}} < \boldsymbol{N}_{\text{gold}}$; *Fully correct hops*	The model executes fewer hops than required, yet all executed reasoning steps are correct and directly correspond to a subset of the required hops. This indicates incomplete but partially correct reasoning.
$\boldsymbol{N}_{\text{model}} < \boldsymbol{N}_{\text{gold}}$; *Partially correct hops*	The model executes fewer reasoning steps than required, omitting essential hops and introducing incorrect hops within the shortened chain. The reasoning is both incomplete and partially incorrect.
$\boldsymbol{N}_{\text{model}} > \boldsymbol{N}_{\text{gold}}$; *Trailing irrelevance*	The model initially executes all required reasoning steps but then continues with additional irrelevant hops. These extra steps occur after completing the required reasoning and reflect the model's extraneous elaboration.
$\boldsymbol{N}_{\text{model}} > \boldsymbol{N}_{\text{gold}}$; *Early irrelevance*	The model introduces irrelevant reasoning steps before or interspersed among the required hops. These interruptions disrupt logical reasoning progression, resulting in confusion, distraction or circular reasoning. The required reasoning steps may be partially addressed or incorrect.
Question misinterpretation	The model misunderstands the original question during its early reasoning steps, often focusing on incorrect entities or setting up the wrong task, leading to fundamentally flawed reasoning.

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. These overlaps revealed that even 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 verbose or circular reasoning within a document, captured by a separate overthinking flag.

3.3 Definitions of Reasoning Categories

Following iterative refinement and extensive pilot annotations, we arrived at a final taxonomy that enabled high inter-annotator agreement. As shown in Figure 4, this version resolved prior ambiguities by enforcing stricter hop semantics and introducing meta-evaluation markers to capture surface-level verbosity independently from structural reasoning failure. Table 1 summarizes our final taxonomy, providing precise operational definitions for each reasoning error category used in our annotation pipeline. These categories, combined with the meta-evaluation markers of overthinking and coverage,

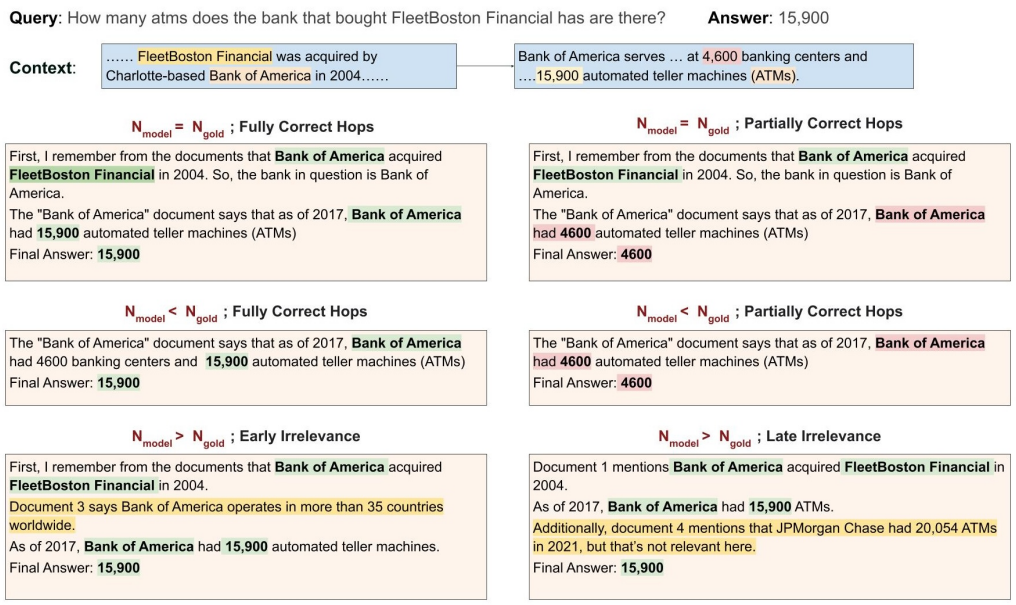


Figure 1: **Examples of Reasoning Error Categories.** Representative outputs illustrating the main error categories in multi-hop reasoning for a single example. The correct entities are highlighted in green, incorrect in red, and irrelevant or extraneous information in yellow.

provide comprehensive coverage of potential reasoning errors, enabling systematic and insightful error diagnosis in multi-hop QA models.

Meta-Evaluation Markers

To further enhance our analytical granularity, we introduced meta-evaluation markers:

Overthinking: This marker captures indicators of cognitive inefficiency in the model's reasoning. It is applied when: 1) the model includes non-essential information from gold documents—such as background details, tangential facts, or calculations—that do not aid in progressing the reasoning chain; and 2) the model demonstrates repetitive or circular behavior, such as repeatedly checking the same entity or relation more than twice.

Coverage: This marker addresses the completeness of source-document utilization, specifically evaluating whether the model successfully retrieves all necessary source documents. Low coverage indicates gaps in retrieval or attention, leading to incomplete reasoning chains or unsupported conclusions.

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. 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. [18], 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 model reasoning across three multi-hop QA datasets of increasing difficulty: 2WikiMultiHopQA [19], which emphasizes structured multi-hop reasoning; HotpotQA [1], which includes distractors and diverse reasoning types like comparisons; and MuSiQue [2], a high-complexity benchmark designed to minimize shortcuts through dense context and sub-question dependencies. Dataset details are provided in Table 4 in the appendix.

Question Types To enable systematic reasoning analysis, we categorize multi-hop questions into five distinct 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. Detailed definitions and illustrative examples for each type are provided in Appendix (Table 3).

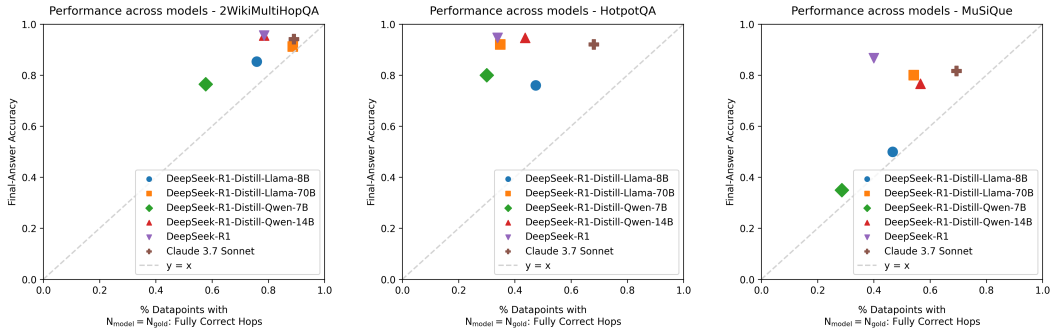
4.1 Annotation process

Figures 5 and 6 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 three 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.
2. **Final answer and meta eval markers:** 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) *Coverage* marker, and (c) *Overthinking* marker.
3. **Reasoning category assignment:** Each response was categorized into one of our predefined reasoning error types (see Section 3.3).

5 Human Evaluation Results

5.1 Reasoning Fidelity and Answer Accuracy



(a) 2Wiki (b) HotpotQA (c) MuSiQue

Figure 2: **Relationship Between Reasoning Fidelity and Answer Accuracy Across Datasets.** Each subplot shows performance on (a) 2Wiki, (b) HotpotQA, and (c) MuSiQue. Points denote models, with the x-axis showing the fraction of fully correct reasoning traces ($N_{\text{model}} = N_{\text{gold}}$) and the y-axis showing answer accuracy. The dotted diagonal ($y = x$) marks perfect alignment; points above it indicate correct answers despite imperfect reasoning.

Figure 2 summarizes model behavior on fully correct hop alignment ($N_{\text{model}} = N_{\text{gold}}$) and final-answer accuracy across datasets. We see that reasoning Fidelity holds in Simpler Tasks but collapses in Complex Chains. Across all datasets, Claude 3.7 achieves the highest accuracy.

High Reasoning Fidelity on 2Wiki: All models perform strongly on the 2Wiki dataset, with the majority of models have around 80% datapoints with $N_{\text{model}} = N_{\text{gold}}$correspondence: fully correct hops, and near-perfect final answer accuracy This confirms that current LMs reliably handle simple multi-hop questions.

Inefficient Reasoning in HotpotQA: Performance on HotpotQA shows the highest concentration of ($N_{\text{model}} > N_{\text{gold}}$) (Figure 7b). While the final-answer accuracy remains high, the presence of semantically dense and distractor-filled paragraphs leads models to over-explore the context, often

beyond the required inference chain. This behavior highlights the limitations of current LMs in maintaining focused reasoning under noisy, multi-document settings.

Intermediate Fidelity and Model-Specific Patterns emerge on the MuSiQue dataset: Larger models demonstrate intermediate reasoning fidelity (45–65%) alongside relatively high answer accuracy. Smaller models exhibit poor performance on both metrics, underscoring difficulties in complex multi-hop contexts. Notably, DeepSeek-R1 shows the greatest divergence, achieving very high answer accuracy despite substantially lower reasoning fidelity.

5.2 Reasoning Patterns Across Models and Datasets

Figure 7 show the distribution of reasoning error types in the MuSiQue, 2Wiki-MultiHopQA, and HotpotQA datasets. Our analysis reveals the following insights:

Claude 3.7 Sonnet Sets the Bar for Stable and Precise Reasoning: Among all evaluated models, Claude 3.7 Sonnet demonstrates the most stable and controlled reasoning behavior. It consistently maintains high rates of fully correct reasoning while keeping all other error types—especially early and trailing irrelevance—significantly lower than both DeepSeek-R1 and the distilled model variants.

Overhopping is the Most Persistent and Systemic Reasoning Failure: Across all datasets and models, overhopping ($N_{\text{model}} > N_{\text{gold}}$ categories in Figure 7) is consistently higher than other errors. This often stems from contextual redundancy or ambiguity, pushing models to over-explore rather than terminate. The Qwen family of models particularly struggles with this issue: frequently displaying early and trailing irrelevance errors—even at larger scales—indicating a proclivity for recall over precise reasoning.

Scaling Models Improves Simple Reasoning but Leaves Complex Errors Unresolved: As shown in Figure 7, increasing model size leads to more examples with fully correct hops (the leftmost bars in each subplot), particularly on simpler tasks like 2Wiki (Figure 7a). However, for more complex datasets such as HotpotQA and MuSiQue (Figure 7b, c), the gains from scaling plateau. Even the largest models still exhibit substantial numbers of early and trailing irrelevance errors (the right-side bars). This persistent error pattern indicates that, while scale enhances basic multi-hop reasoning, it does not fully resolve deeper reasoning challenges in complex/distractor-heavy settings.

Deepseek-R1 Distilled Models Rival the Deepseek-R1 Counterpart in Multi-hop Tasks: On both simple and moderately complex datasets, distilled LLaMA variants show strong reasoning alignment. The LLaMA 70B variant performs almost similarly or even better than the original Deepseek-R1 model.

5.3 Relationship Between Reasoning Errors and Final Answer Correctness

Figure 8 examines the relationship between reasoning trace quality and final answer correctness across datasets.

Answer Correctness is Sensitive to Missing Hops: Looking at the "Partially Correct Hops" bars in all three panels of Figure 8, we see that incomplete reasoning rarely yields a correct answer. This confirms that failure to cover all necessary facts, even in part, is a definitive bottleneck in LMs' reasoning chains.

Smaller Models are More Fragile to Reasoning Errors: As shown in Figure 8 across all datasets, smaller models such as LLaMA-8B and Qwen-7B exhibit a higher propensity for reasoning errors to cascade into incorrect final answers. In contrast, larger models like DeepSeek-R1 and Claude 3 Sonnet demonstrate greater robustness, with fewer incorrect answers arising from these types of reasoning errors.

Early Irrelevance is More Detrimental than Trailing Irrelevance: In every panel of Figure 8, the "Early Irrelevance" category shows that "Answer Incorrect" bars are higher compared to that corresponding to "Trailing Irrelevance." This suggests that irrelevant reasoning steps introduced early in the chain are more disruptiveto the model's final answer.

5.4 Overthinking Trends and Their Impact

We systematically examine the prevalence and impact of overthinking across different models and datasets, highlighting how this phenomenon influences overall model performance and error rates.

Overthinking Surges in Complex Reasoning Tasks: As shown in the MuSiQue results (see Figure 7c and Table 5), overthinking rises markedly across all models, with rates ranging from 36.7% to 61.7%. Notably, DeepSeek-R1-Distill-Qwen-7B reaches the highest overthinking rate of 61.7%, while even advanced models such as Claude 3 Sonnet and DeepSeek-R1 exhibit elevated rates. This trend suggests that task complexity, rather than model scale, is the primary driver of overthinking.

Overthinking is a Systematic Source of Incorrect Answers: A significant portion of incorrect answers are accompanied by overthinking, especially in MuSiQue (see Figure 9a). Although HotpotQA and 2Wiki contain fewer errors labeled as overthinking (see Table 5), when overthinking does occur, it almost always results in incorrect answers (see shaded bars in Figure 9c and Figure 9b). This finding suggests that the negative impact of overthinking is not just limited to complex datasets but also arises from the logical incoherence it introduces, irrespective of task difficulty. Overthinking is not merely harmless elaboration, but a systematic driver of reasoning collapse and failure to reach a final answer.

5.5 Distribution across Question types

Figure 10 shows the distribution of reasoning error types across question categories for all the models. We observe the following trends across different question types:

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 Qwen-7B and LLaMA-8B perform well, while Claude 3.7 Sonnet and Qwen-14B make no errors. These questions often contain explicit reference to entities or co-occurrence patterns that mirror the pre-training distribution of the model, allowing models to resolve them through recognition of patterns at the surface level rather than deep reasoning.

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 (Qwen-7B, LLaMA-8B, DeepSeek-R1) show many partially correct chains, even with correct hop counts. Claude and 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, misinterpretation, and underhopping. Qwen-7B answers only 10% correctly, with 30% misinterpretation. Even DeepSeek-R1 shows 37% trailing irrelevance. Only Claude and LLaMA-70B manage modest control (50–55% correct), highlighting the inherent difficulty of inference.

Inference and Compositional Tasks Drive Overthinking: Overhopping is most common in inference questions, reaching 70% in Qwen-7B, 65% in LLaMA-8B, and 60% in DeepSeek-R1 and Qwen-14B. Lack of clear stopping cues leads models to overgenerate. Bridge questions show minimal overhopping (<20%) due to their bounded structure.

5.6 Hop-wise Error Distribution

To better understand how reasoning evolves across multi-hop inference chains, we analyze the distribution of reasoning errors at the hop level. Figure 11 illustrates these trends for all models.

Larger Models Are More Stable Across Hop Counts: As the number of required reasoning steps increases, most models exhibit a clear drop in fully correct reasoning ($N_{\text{model}} = N_{\text{gold}}$). For example, in Figure 11a (left panel), DeepSeek-R1-Distill-Llama-8B achieves 53% accuracy 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 (Figure 11b, right panel) and Claude 3.7 Sonnet (Figure 11d) show much greater stability across hop lengths, maintaining relatively consistent performance even as reasoning depth increases. This suggests these models have a stronger capacity to follow and complete longer reasoning chains without deviation.

Overhopping Is a Major Error Source in 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 (Figure 11a), where 73% of 4-hop examples are categorized as early irrelevance. Both Claude 3.7 Sonnet and DeepSeek-R1-Distill-Qwen-14B (Figure 11b and d) show 45% early irrelevance at 4 hops. These results indicate that, in more complex tasks, models frequently continue reasoning beyond what is necessary, retrieving irrelevant information.

Shallow Collapse in Qwen-7B, Depth Limitations in Claude 3.7 Sonnet: Qwen-7B (Figure 11a) shows signs of partial reasoning at 3 hops but collapses almost entirely into early irrelevance (73%) at 4 hops, abandoning intermediate reasoning strategies. This suggests that, under high reasoning load, smaller models tend to default to over-retrieval. In contrast, Claude 3.7 Sonnet (Figure 11d) maintains strong performance up to 3 hops but shows a spike in early irrelevance (45%) at 4 hops. Even advanced models, therefore, encounter depth calibration issues, struggling to determine when to stop in extended reasoning chains.

6 Automated Evaluation Results

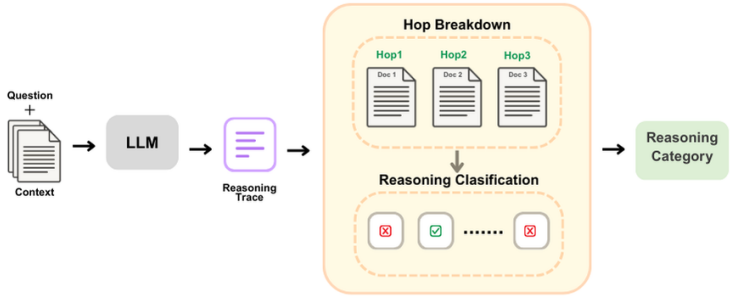


Figure 3: **Two-Step LLM-Assisted Evaluation Workflow.** A high-level overview of the two-step decomposition that improves annotation accuracy and consistency for complex multi-hop reasoning.

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.

Consistent with findings for multi-step judging process in [20, 12], we adopt a two-step annotation process as illustrated in Figure 3. 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 [21].

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 2, 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 like DeepSeek-R1 and LLaMA 70B exhibit notably higher agreement rates, particularly on simpler datasets like 2WIKI, achieving above 90%. Conversely, 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.

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 advanced refinement of Judge LLM instructions. 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.

Table 2: LLM-as-a-Judge Agreement (%) with Human Annotations Across Models and Datasets. We find that across models LlaMa 70B and Claude 3.7 has highest agreement scores.

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

7 Conclusion and Limitations

We introduce a hop-based diagnostic framework for multi-hop QA that captures reasoning fidelity through fine-grained error categories and meta-markers for coverage and overthinking. Analysis of six LMs across three datasets reveals high fidelity in simple settings but persistent overhopping, misinterpretation, and synthesis failures in complex and distractor-rich tasks. Our two-step LLM-as-a-Judge method achieves up to 92% agreement with humans on simpler datasets while cutting evaluation time by 20x, although challenges remain for nuanced reasoning. These findings call for evaluation and training strategies that bridge the gap between correct answers and reasoning that is both efficient and faithful for truly reliable multi-hop QA systems.

This paper was limited to text only multi-hop analysis, as other modalities of search and synthesis using AI-bots are becoming popular, building a similar repositiory of failure modes will be pivotal to finding gaps in reasoning models and fixing them.

References

[1] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. *CoRR*, 2018. URL http://arxiv.org/abs/1809.09600.

[2] Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. MuSiQue: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:533–550, 2022. doi: 10.1162/tacl_a_00475. URL https://aclanthology.org/2022.tacl-1.31/.

[3] Jian Wu, Linyi Yang, Zhen Wang, Manabu Okumura, and Yue Zhang. CofCA: A STEP-WISE counterfactual multi-hop QA benchmark. In *The Thirteenth International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=q2DmkZ1wVe.

[4] Mansi Sakarvadia. Towards interpreting language models: A case study in multi-hop reasoning, 2024. URL https://arxiv.org/abs/2411.05037.

[5] Chirag Agarwal, Sree Harsha Tanneru, and Himabindu Lakkaraju. Faithfulness vs. plausibility: On the (un)reliability of explanations from large language models, 2024. URL https://arxiv.org/abs/2402.04614.

[6] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed Chi, Quoc Le, and Denny Zhou. Chain-of-thought prompting elicits reasoning in large language models, 2023. URL https://arxiv.org/abs/2201.11903.

[7] Guangsheng Bao, Hongbo Zhang, Cunxiang Wang, Linyi Yang, and Yue Zhang. How likely do llms with cot mimic human reasoning?, 2024. URL https://arxiv.org/abs/2402.16048.

[8] Parshin Shojaee, Iman Mirzadeh, Keivan Alizadeh, Maxwell Horton, Samy Bengio, and Mehrdad Farajtabar. The illusion of thinking: Understanding the strengths and limitations of reasoning models via the lens of problem complexity, 2025. URL https://arxiv.org/abs/2506.06941.

[9] Ai Ishii, Naoya Inoue, Hisami Suzuki, and Satoshi Sekine. Analysis of LLM‘s “spurious” correct answers using evidence information of multi-hop QA datasets. In Russa Biswas, Lucie-Aimée Kaffee, Oshin Agarwal, Pasquale Minervini, Sameer Singh, and Gerard de Melo, editors, *Proceedings of the 1st Workshop on Knowledge Graphs and Large Language Models (KaLLM 2024)*, pages 24–34, Bangkok, Thailand, August 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.kallm-1.3. URL https://aclanthology.org/2024.kallm-1.3/.

[10] 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, Saurav Kadavath, Shannon Yang, Thomas Henighan, Timothy Maxwell, Timothy Telleen-Lawton, Tristan Hume, Zac Hatfield-Dodds, Jared Kaplan, Jan Brauner, Samuel R. Bowman, and Ethan Perez. Measuring faithfulness in chain-of-thought reasoning, 2023. URL https://arxiv.org/abs/2307.13702.

[11] Xiaoyuan Li, Wenjie Wang, Moxin Li, Junrong Guo, Yang Zhang, and Fuli Feng. Evaluating mathematical reasoning of large language models: A focus on error identification and correction, 2024. URL https://arxiv.org/abs/2406.00755.

[12] Chujie Zheng, Zhenru Zhang, Beichen Zhang, Runji Lin, Keming Lu, Bowen Yu, Dayiheng Liu, Jingren Zhou, and Junyang Lin. Processbench: Identifying process errors in mathematical reasoning, 2024. URL https://arxiv.org/abs/2412.06559.

[13] Sagnik Mukherjee, Abhinav Chinta, Takyoung Kim, Tarun Anoop Sharma, and Dilek Hakkani-Tür. Premise-augmented reasoning chains improve error identification in math reasoning with llms, 2025. URL https://arxiv.org/abs/2502.02362.

[14] 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. Evaluating llms at detecting errors in llm responses, 2024. URL https://arxiv.org/abs/2404.03602.

[15] Yongqi Tong, Dawei Li, Sizhe Wang, Yujia Wang, Fei Teng, and Jingbo Shang. Can llms learn from previous mistakes? investigating llms' errors to boost for reasoning, 2024. URL https://arxiv.org/abs/2403.20046.

[16] Hamed Mahdavi, Alireza Hashemi, Majid Daliri, Pegah Mohammadipour, Alireza Farhadi, Samira Malek, Yekta Yazdanifard, Amir Khasahmadi, and Vasant Honavar. Brains vs. bytes: Evaluating llm proficiency in olympiad mathematics, 2025. URL https://arxiv.org/abs/2504.01995.

[17] 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. Errorradar: Benchmarking complex mathematical reasoning of multimodal large language models via error detection, 2024. URL https://arxiv.org/abs/2410.04509.

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

[19] Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps, 2020. URL https://arxiv.org/abs/2011.01060.

[20] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. Self-consistency improves chain of thought reasoning in language models, 2023. URL https://arxiv.org/abs/2203.11171.

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

A Appendix

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.

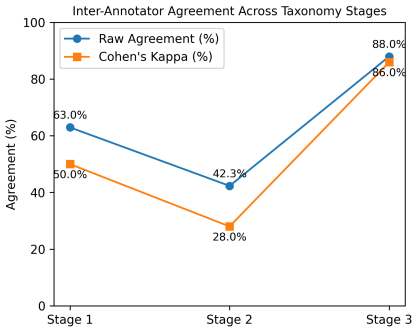


Figure 4: **Improvement in Inter-Annotator Agreement Across Refinement Stages.** Raw agreement and Cohen's kappa both increase substantially as the reasoning error taxonomy evolves from loosely defined to formally structured categories, with the highest agreement achieved after Stage 3 refinements.

Table 3: Examples of different question types with highlighted entities

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?

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

Dataset Details

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

- **2WikiMultiHopQA** [19]: 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** [1]: 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** [2]: Presents 20-document contexts per question. Constructed to reduce shortcut-based reasoning by enforcing sub-question dependencies and including challenging unanswerable distractors.

Overthinking Rates

Table 5: **Overthinking Rates by Model and Dataset.** Percentage of answers with Overthinking for each model on the 2Wiki-MultiHopQA, HotpotQA, and MuSiQue datasets. The results highlight the substantial increase in overthinking in more complex MuSiQue dataset.

Model	2Wiki-MultiHopQA	HotpotQA	MuSiQue
DeepSeek-R1-Distill-Llama-8B	41.2%	29.3%	48.3%
DeepSeek-R1-Distill-Llama-70B	19.1%	12.0%	41.7%
DeepSeek-R1-Distill-Qwen-7B	26.5%	41.3%	61.7%
DeepSeek-R1-Distill-Qwen-14B	30.9%	28.0%	50.0%
DeepSeek-R1	27.9%	18.7%	53.3%
Claude 3.7 Sonnet	22.1%	22.7%	36.7%

B Additional Figures

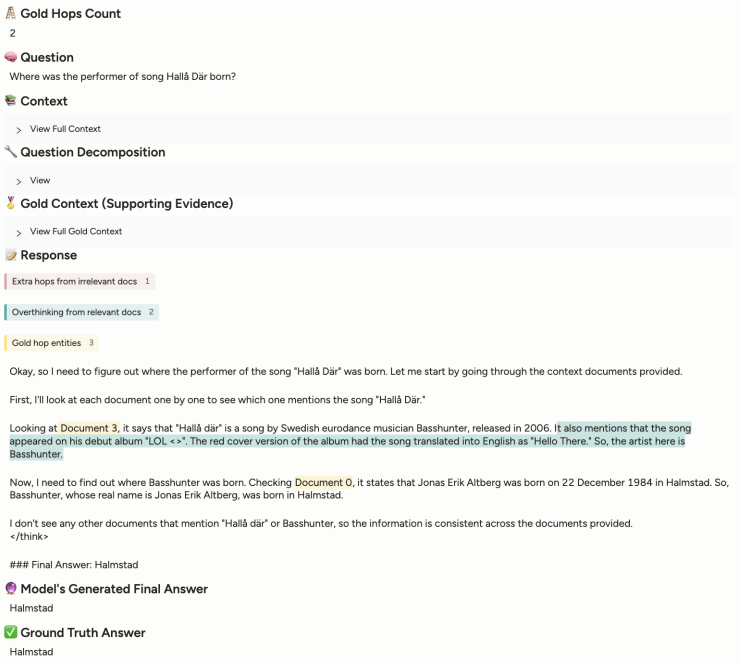


Figure 5: Example of Input given to annotators

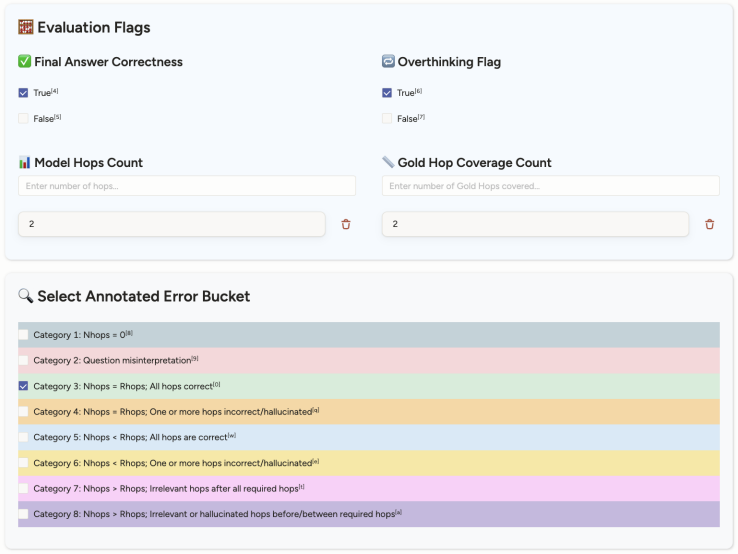
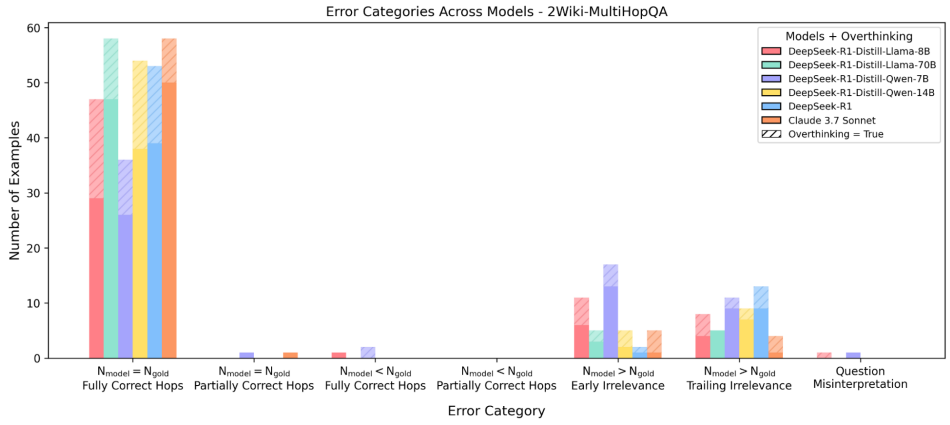
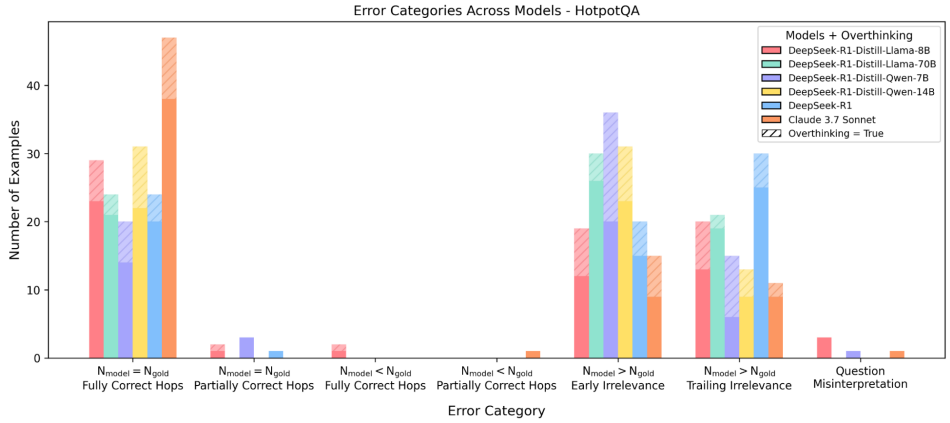


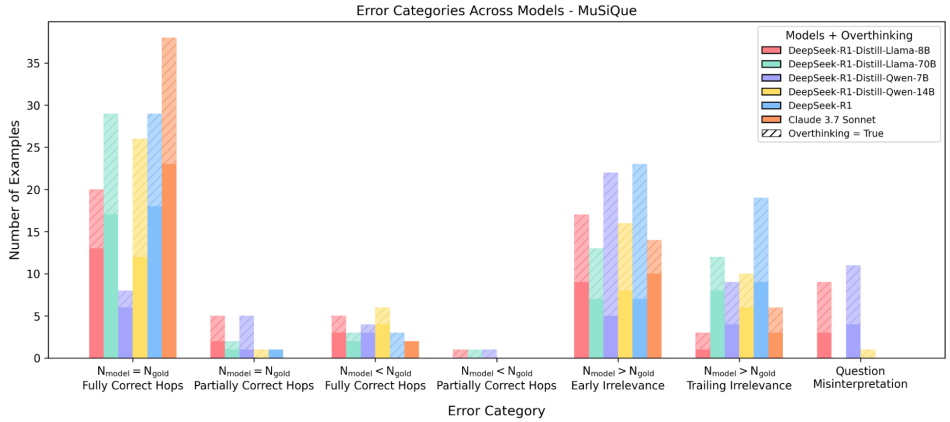
Figure 6: Example of Output labeled by the human annotators



(a) 2Wiki

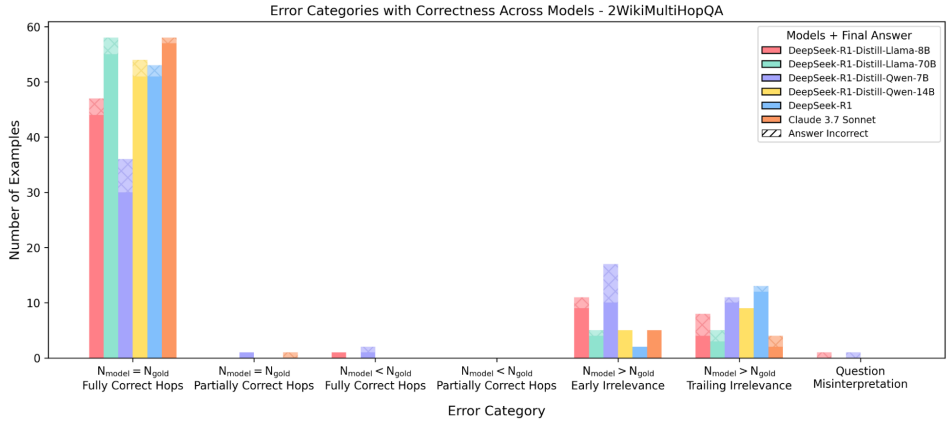


(b) HotpotQA

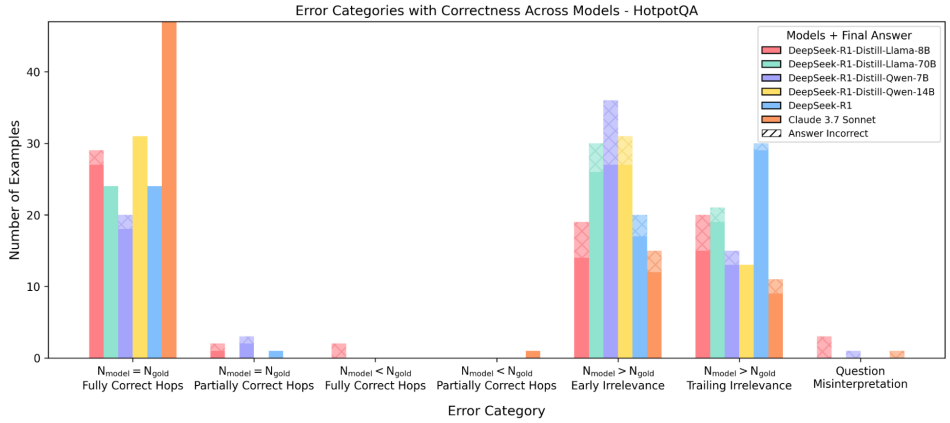


(c) MuSiQue

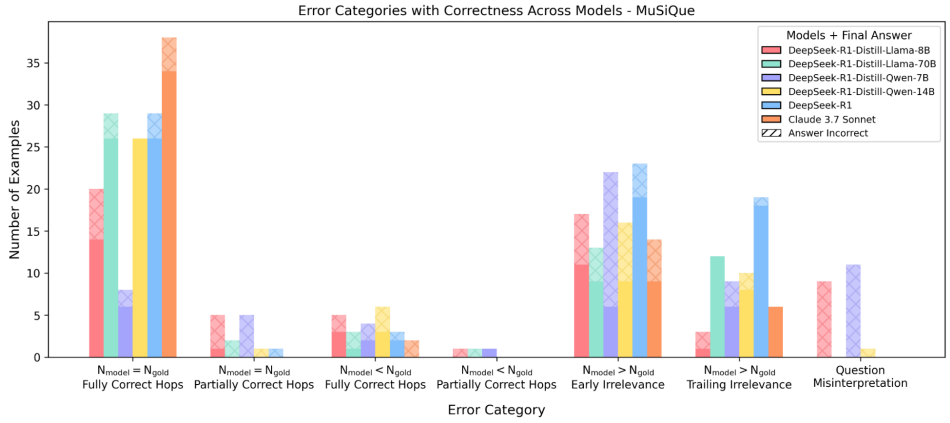
Figure 7: Distribution of reasoning error types across datasets. (a) 2WIKI, (b) HOTPOTQA, (c) MUSIQUE.



(a) 2Wiki

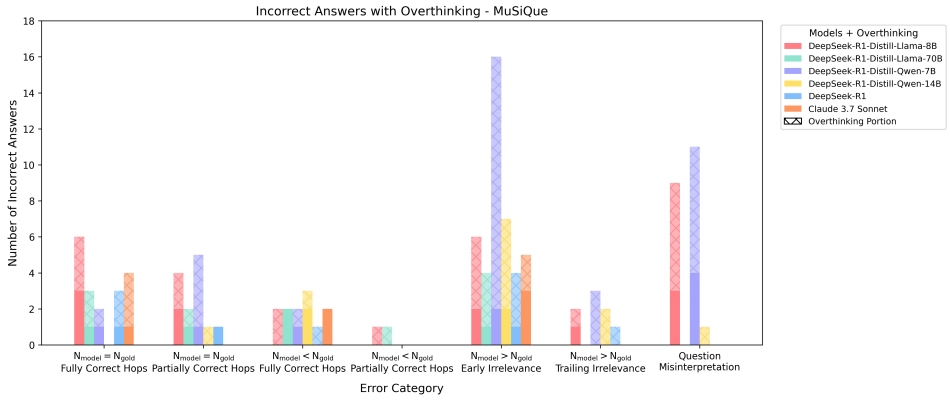


(b) HotpotQA

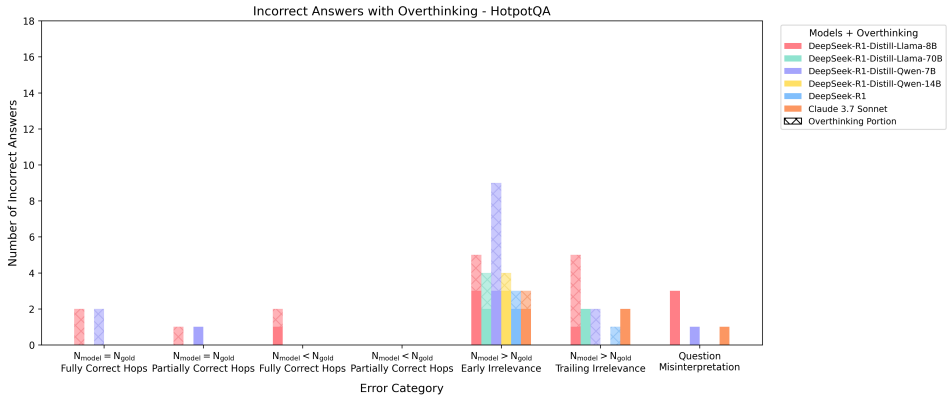


(c) MuSiQue

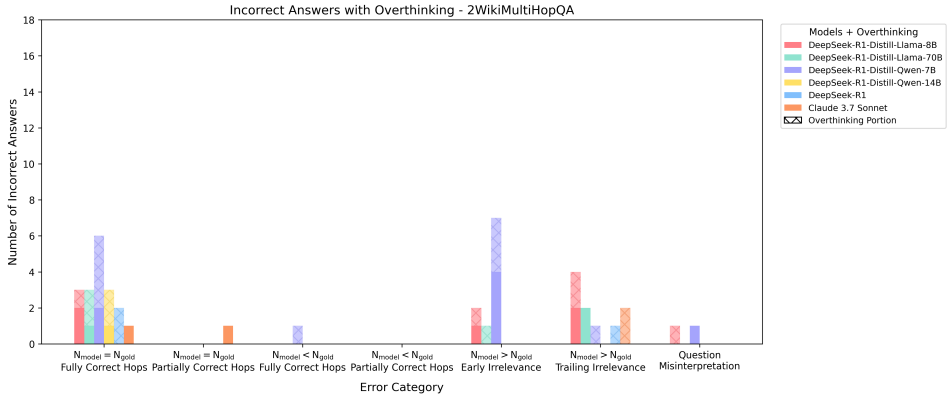
Figure 8: Answer correctness breakdown by reasoning category across datasets. (a) 2WIKI, (b) HOTPOTQA, (c) MUSIQUE.



(a) MuSiQue

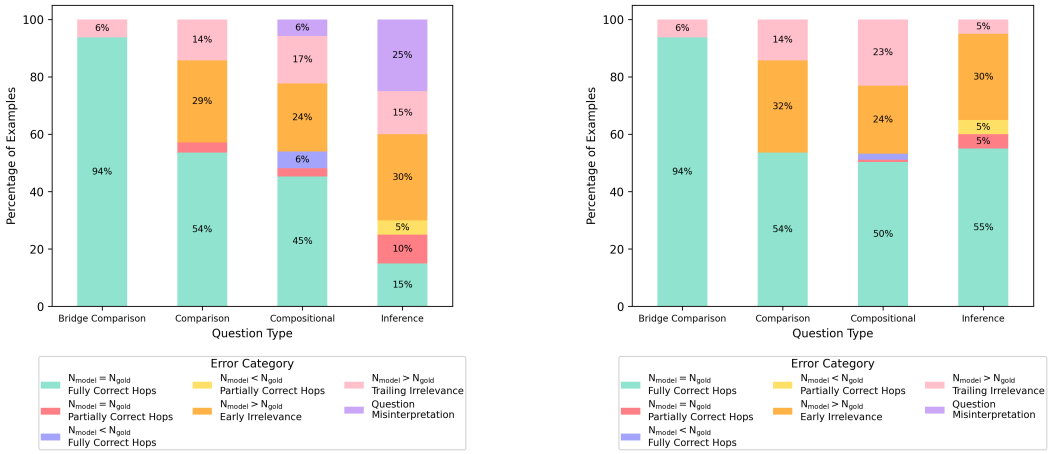


(b) HotpotQA



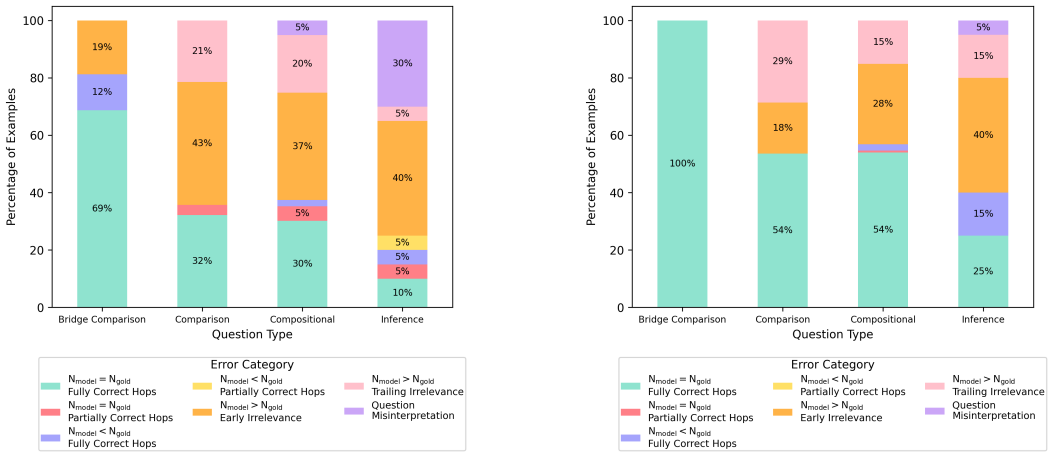
(c) 2Wiki

Figure 9: Overthinking Trends with Answer Incorrectness across Datasets. (a) MuSiQue, (b) HotpotQA, and (c) 2Wiki.



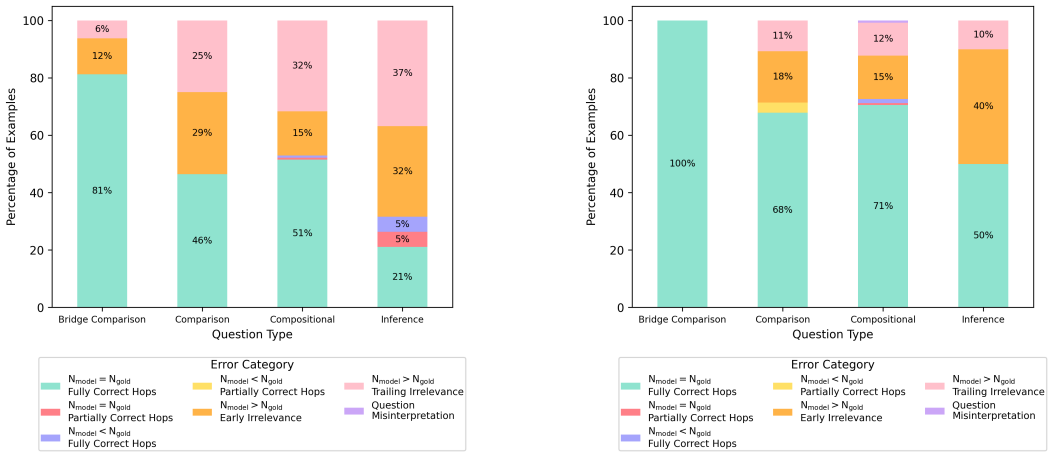
(a) LLaMA-8B (Distill)

(b) LLaMA-70B (Distill)



(c) Qwen-7B (Distill)

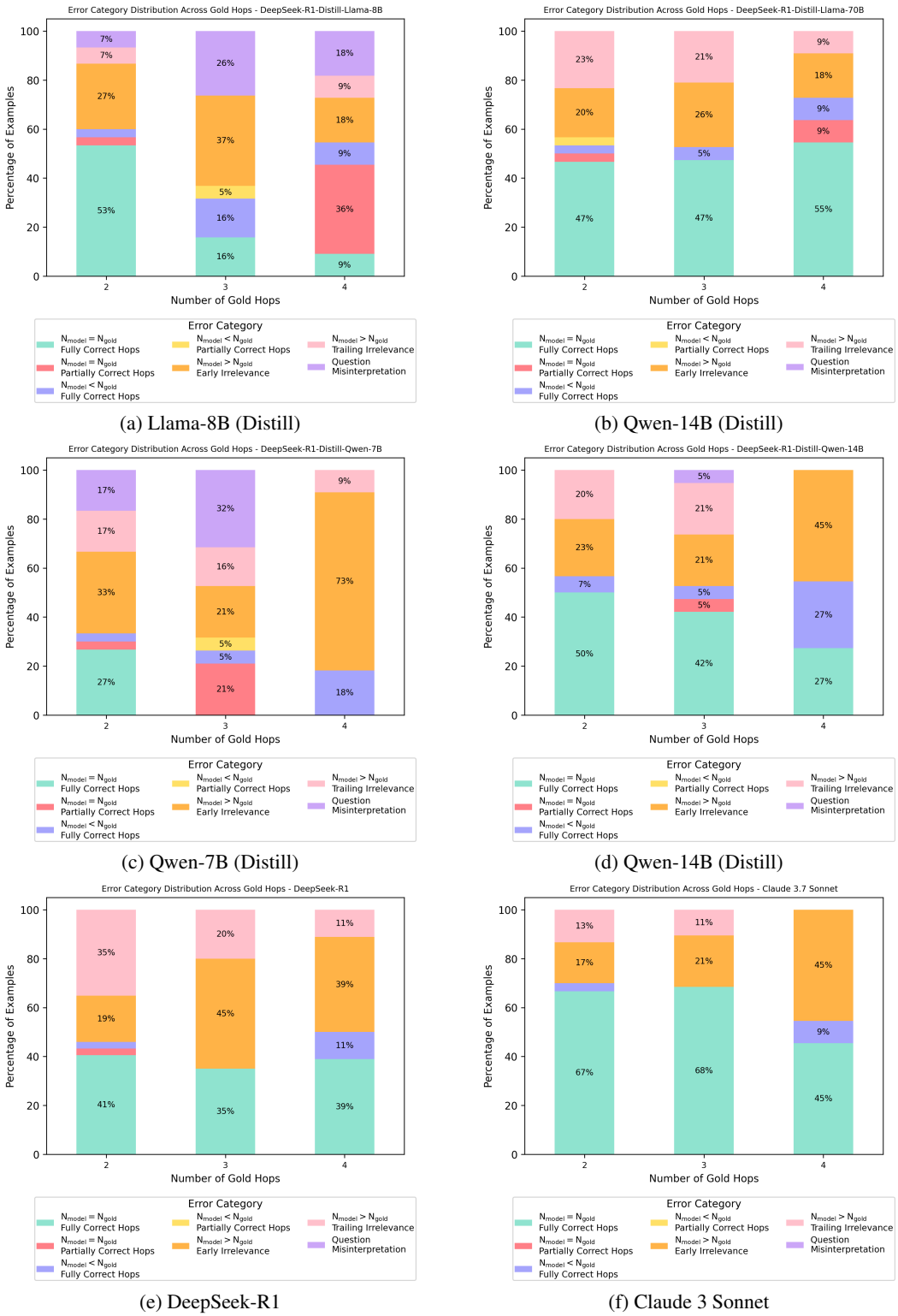
(d) Qwen-14B (Distill)



(e) DeepSeek-R1

(f) Claude 3 Sonnet

Figure 10: Distribution of reasoning error types across question types for six models. Each subfigure shows model-specific trends in how question type impacts reasoning errors.



(a) Llama-8B (Distill)

(b) Qwen-14B (Distill)

(c) Qwen-7B (Distill)

(d) Qwen-14B (Distill)

(e) DeepSeek-R1

(f) Claude 3 Sonnet

Figure 11: Hop-wise distribution of reasoning errors on MuSiQue for four models. Subplots (a)–(f) show how models vary in reasoning step correctness and overhopping behavior. Results highlight the decline in fully correct reasoning with greater hop count, and the increasing prevalence of overhopping errors on harder questions.

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 3. 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 models 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"
}
```

NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes] , [No] , or [NA] .
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

The checklist answers are an integral part of your paper submission. They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes] " is generally preferable to "[No] ", it is perfectly acceptable to answer "[No] " provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No] " or "[NA] " is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading "NeurIPS Paper Checklist"**,
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers**.

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: Yes, the claims are summarized in "Contributions" within the Introduction are appropriately substantiated throughout the paper and additional details and figures have been added to the Appendix.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

Justification: Paper notes lower judge agreement on MuSiQue and need for human oversight; challenges remain for nuanced reasoning (Sec. 5.2 and Section 7: Conclusion and Limitations section)

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA] .

Justification: No formal theorems/proofs; work is applied where we develop a detail taxonomy of reasoning model failure modes and validate the occurrence of such failures with detailed experiments across proprietary and open source reasoning models like DeepSeek, Claude etc.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: Yes, the paper highlights the prompt for scaling annotation of failure patterns, provides details of not just taxonomy, but iteration and concepts, as well as explanations to all reasoning model failures in Appendix C.1.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [NA]

Justification: aper does not include solutions that require code. Annotation pipeline is committed to Github and will be available if requested.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.

- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: Yes. Inference settings (e.g., temperature 0.6 for DeepSeek, 0.0 for Claude), datasets, question types, sample sizes, and annotation steps are specified (Sec. 3.1).

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Detailed plots showing distributions of failure across the reasoning models and datasets have been added in Results section as well as Appendix B. With an additional page, if the paper is accepted, we plan to add more box plots showing distribution of annotations.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.
   - It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
   - For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
   - If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [NA]

Justification: This paper doesn't involve training a model for solution, rather just relies on model inferencing (API calls not GPU dependent) to generate responses and traces to analyze/manually annotate them

Guidelines:

- The answer NA means that the paper does not include experiments.
- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

Answer: [Yes]

Justification: We use public QA datasets and model outputs; no personal/sensitive data; analysis/annotation only (Sec. 3 Datasets).

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [Yes]

Justification: We highlight and argue for fidelity/robustness benefits for reliable systems throughout the paper through identifying failure patterns and finding the root cause for remediation (Results; Conclusion)

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.

- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [No]

    Justification: No new high-risk models/datasets release.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All papers and datasets are appropriately cited

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
    - If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

Justification: Error categories, conceptualization, and taxonomy is being released. With the camera ready version, raw annotations for all data points will also be open sourced.

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

Answer: [NA]

Justification: Human annotation is described as an internal labeling pipeline; no crowdsourcing study or participant research reported

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
- According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: No human-subjects study requiring IRB is described.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer: [Yes]

Justification: We used LLM-as-a-Judge in our experiments to scale up the annotation of failure modes and also correlate it with human annotators/judges. The details of this setup to use a LLM-judge for annotation reasoning model failures for multi-hop conversational search is central to the paper, and has been documented (two-step hop breakdown + classification, model named, guidance provided) in Section 5.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.